

GENERIC MULTI-ATTRIBUTE ANALYSIS (GMAA) USER'S GUIDE

Antonio Jiménez, Sixto Ríos-Insua and Alfonso Mateos

Technical University of Madrid

School of Computer Science

Decision Analysis and Statistics Group

<http://www.dia.fi.upm.es/grupos/dasg/index.htm>

Facultad de Informática, Campus de Montegancedo S/N, Boadilla del
Monte 28660, Madrid, SPAIN

Contents

1	INTRODUCTION	3
1.1	Installing the Software	4
1.2	General appearance, main menu and toolbar	6
1.3	Input/Output Files	12
2	BUILDING AN OBJECTIVES HIERARCHY	19
3	ALTERNATIVES PROCESSING	22
4	QUANTIFYING PREFERENCES	30
4.1	Assessment of component utilities	30
4.1.1	Imprecise Piecewise Linear Utility Functions (Direct Assignment)	32
4.1.2	Imprecise Piecewise Linear Utility Functions (CE-Method/PE- Method)	34
4.1.3	Imprecise Utilities for Discrete Attribute Values	48
4.1.4	Direct Assignment using Subjective Scales	50
4.2	Weight Elicitation	51
4.2.1	Weight Elicitation Based on Trade-offs	52
4.2.2	Direct Assignment	55
5	EVALUATION OF ALTERNATIVES	57
6	SENSITIVITY ANALYSIS	66
6.1	Dominance and Potential Optimality	69
6.2	Weight Stability Analysis	71
6.3	Simulation Techniques for SA	74
6.3.1	Random Weights	74
6.3.2	Rank Order Weights	76
6.3.3	Response Distribution Weights	77
7	HELP, PRINT AND PRINT PREVIEW	78

1 INTRODUCTION

Many complex decision-making problems have multiple objectives. These multiple objectives may be conflicting in the sense that, once dominated alternatives have been discarded, further achievement in terms of one objective can only occur at the expense of some achievement of another objective. Therefore, preference trade-offs between different degrees of achievement of one objective or another must be taken into account by the decision maker (DM). Also, real problems are usually plagued by uncertainty. One cannot predict with certainty the consequences of each alternative under consideration. Formal analysis is required because it is very difficult to consider the above complexities informally in the mind.

The goal of decision analysis (DA) is to structure and simplify the task of making hard decisions as well and as easily as the nature of decision permits (Belton, 1990). DA is especially concerned with multiple conflicting objectives.

DA is developed on the assumption that the alternatives will appeal to the expert depending on:

- the likelihood of the possible consequences of each alternative,
- the expert's preferences concerning the possible consequences.

What makes DA unique is the form in which these factors are quantified and formally incorporated into problem analysis. Existing information, collected data, models and professional judgements are used to quantify the likelihood of a range of consequences. Utility theory is used to quantify preferences.

Let us divide DA into four steps:

- *Structuring the problem*, which includes building a value hierarchy and specifying objectives and attributes. This step is explained in section 2.
- *Identifying the feasible alternatives/strategies*, their impact or consequences and uncertainty (if necessary). We present alternatives processing, including adding an alternative, modifying the alternative consequences/names or removing an alternative in section 3.

- *Quantifying preferences*, which includes the assessment of the component utilities for the attributes, weights assignment methods (direct assignment and weights elicitation based on trade-offs) and subjective scales (see section 4).
- *The evaluation of alternatives*. The way the alternatives are classified/ranked and other useful information for the DM are explained in section 5.
- *Sensitivity Analysis*. Several types of sensitivity analysis are presented in section 6.

The DM must follow the above-mentioned steps to construct a workspace which represents a multi-attribute decision problem.

Before explaining the different steps of DA, let us install the software and take a look at the system, its appearance, main menu, toolbar and other interesting features.

1.1 Installing the Software

Generic Multi-Attribute Analysis can be installed from diskettes or CD-ROM depending on which is supplied in your package.

To install from CD-ROM:

- 1 Ensure your CD-ROM drive is installed and working properly.
- 2 Insert the installation CD-ROM into your CD-ROM drive. The CD-ROM does not support Windows **AutoPlay** mode and does not start running automatically. So, double-click the **My Computer** icon on your Windows desktop and right-click the CD-ROM drive icon.
- 3 Double-click **setup.exe** and follow the instructions on the screen to complete the installation.

To install from diskettes:

- 1 Insert the first installation diskette into your floppy disk drive.
- 2 Click the **Start** button, and then click **Run**.
- 3 In the Run dialog box, type **A:\SETUP**, where A is the drive into which you have inserted the diskette.

- 4 Click the **OK** button and follow the instructions on the screen to complete the installation.

In both installations, the system creates a folder in the selected directory, see Figure 1.1.

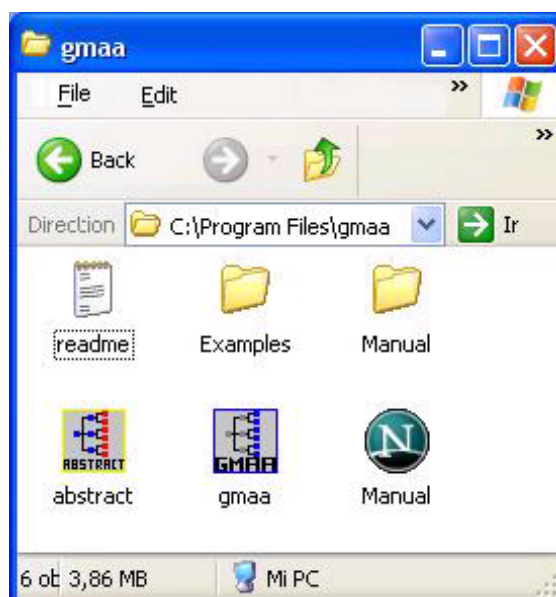


Figure 1.1. Generic Multi-Attribute Analysis application files

Double-click the **gmaa.exe** icon to execute the application and the **manual.html** icon to view the application manual or user's guide. You can also execute the application or view the user's guide by clicking the **Start** button, pointing to **Programs**, and then clicking **Generic Multi-Attribute Analysis**, see Figure 1.2.

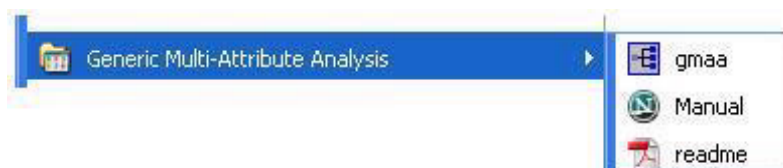



Figure 1.2. Executing the application and viewing the user's guide and readme file

Using the Windows Uninstall feature, you can remove applications cleanly and then reinstall them to correct problems, change configurations or make version updates.

To uninstall the application:

- 
- The screenshot shows the 'Program and Features' window in Windows 7. The 'GMAA' program is selected in the list. The details for this program are shown on the right: Size is 7.45MB and it is used 'Occasionally'. Below the details, there is a text instruction: 'To modify or remove this program from your computer, click on Modify or Remove'. At the bottom right, there are two buttons: 'Modify' and 'Remove'.
- | Name | Size | Used |
|------|--------|--------------|
| GMAA | 7.45MB | Occasionally |
- To modify or remove this program from your computer, click on Modify or Remove
- Modify Remove

1.2 General appearance, main menu and toolbar

GMAA (Generic Multi-Attribute Analysis) -

WorkSpace File View Sensitivity Analysis Window Help

Using the **WorkSpace** menu, see Figure 1.5, we can create, load, save, or close a workspace. All the information related to the problem (i.e., the objectives hierarchy, weights and component utilities information and alternative consequences) is saved in a workspace file. If we select the **New WorkSpace** option, we have to follow the steps defined in DA to represent the multi-attribute decision problem.

We can also print the information on alternatives and attributes and the alternatives ranking using the **Print**, **Print Preview** and **Print Setup** options. The **Save Results to File** option saves the alternatives ranking to file, taking into account the current appearance of the tree, alternative consequences and component utilities or

subjective scales and weights. Finally, we can quit the application by selecting the **Exit** option in this menu.

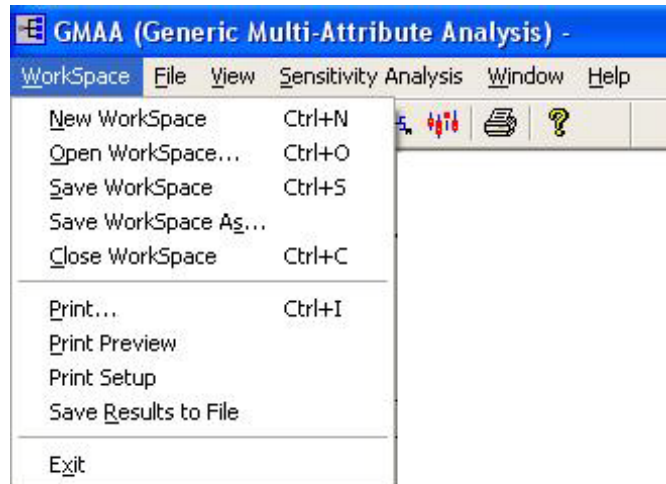


Figure 1.5. **WorkSpace** options

If we have already loaded a workspace or built a new objectives hierarchy, the component utilities in the attributes, average normalized weights and normalized weight intervals or alternative consequences can be saved or loaded using the **File** menu, see Figure 1.6. If we load one of these files, the application will check consistency with the current workspace. For example, when we load a component utilities file the number of leaves and names in the file must be the same as in the current workspace.

The **View** menu (Figure 1.7) is very useful for DMs. Using this menu, we can view all the information related to the workspace. Using the **View Component Utilities** or the **View Alt. Consequences**, all the component utility functions and all the alternatives are displayed, respectively. The **View Alt. Classification** option is used to view the alternatives classification with the current weights, component utilities and alternative consequences.

The **Sensitivity Analysis** menu, see Figure 1.8, shows three options, the **Weight Stability Intervals** option, the **Dominance/Potential Optimality** option and the **Simulation Techniques for SA** option, which are explained in section 6.

Several workspaces can be loaded or created simultaneously. The **Window** menu is used to switch from one to another, i.e., to select the active workspace. The **Window**

menu options are the names of the workspaces, and we just have to select one to make a specific workspace active. We can find out which workspace is active by just reading the name in the bottom right-hand corner of the window, see Figure 1.9.



Figure 1.6. **File** options

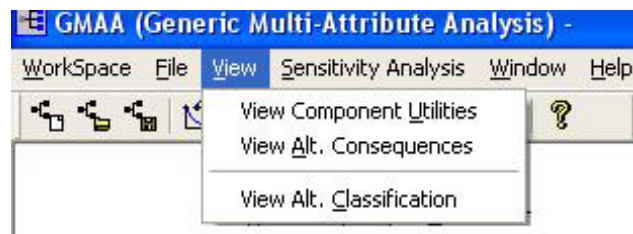


Figure 1.7. **View** options



Figure 1.8. **Sensitivity Analysis** options

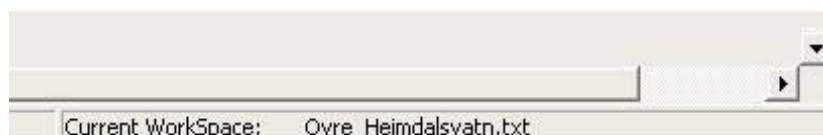


Figure 1.9. The current workspace

Finally, information about individuals and institutions involved in the development of the GMAA module is shown by selecting the **About** option in the **Help** menu, see Figure 1.10.

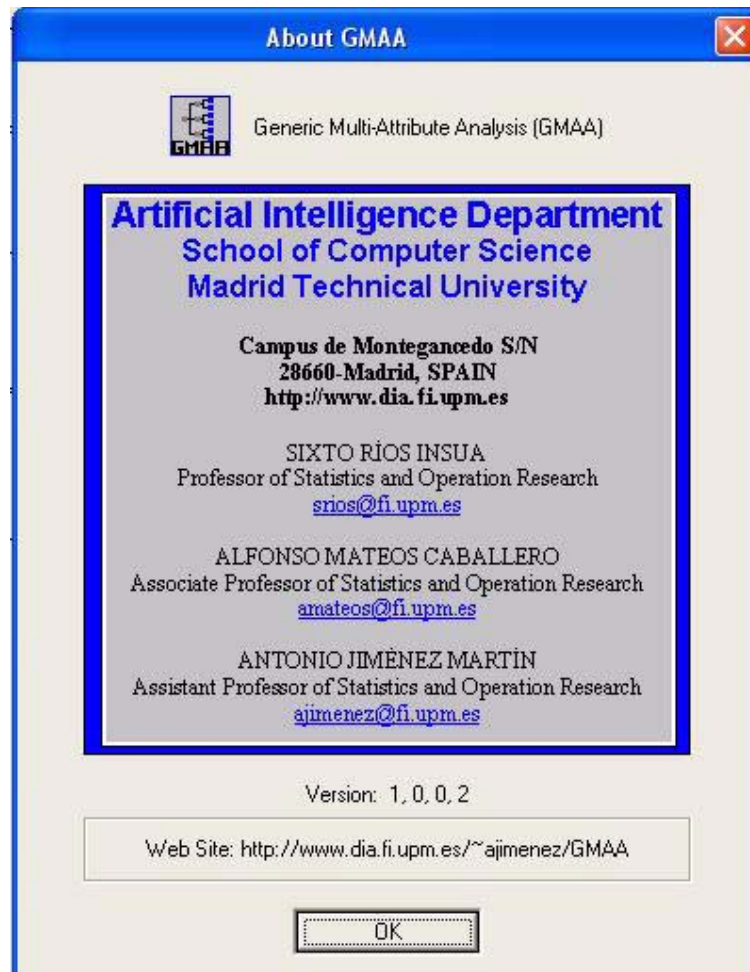


Figure 1.10. **About GMAA**

The appearance of the toolbar is as follows:



Figure 1.11. The program toolbar

The first three buttons are related to workspaces. The first one can be used to create a new workspace, the second one to open an existing workspace and the last one to save the current workspace.

The next three buttons are equivalent to the **View** options. They can be used to view the component utilities, the alternative consequences and the alternatives classification, respectively.

The next three buttons are related to Sensitivity Analysis. The first one is equivalent to the **Weights Stability Intervals** option, the second to the **Dominance / Potential Optimality** option and the third to the **Simulation Techniques for SA** option in the **Sensitivity Analysis** menu.

The **Print** button is equivalent to the **Print** option in the **WorkSpace** menu. Finally, the last button is equivalent to the **About GMAA** option in the **Help** menu.

When we start up the program, the window illustrated in Figure 1.12 is displayed. At this point no workspace has been loaded, so many menu options are not activated. The DM has to load an existing workspace or create a new one following the DA steps. In this user's manual, an example workspace is used to explain the system. This specimen workspace is called *Øvre Heimdalsvatn*, and its appearance is as shown in Figure 1.13.

The aim of this example is to identify optimal remedial strategies for the restoration of the *Øvre Heimdalsvatn* Lake (Oppland Norway) aquatic ecosystem, contaminated by radionuclides (see Gallego *et al.* 1998, Ríos Insua *et al.* 2000 and Jiménez *et al.* 2002). This problem has been studied in depth in the European Projects in which we have participated: **MOIRA** (A **MO**del-based computerised system for management support to **I**dentify optimal **R**emedial strategies for restoring radionuclide contaminated **A**quatic ecosystem and drainage areas, 1996-1998), **COMETES** (Implementing **CO**mputerized **MET**hologies to **E**valuate the effectiveness of countermeasures for restoring radionuclide contaminated fresh water eco**S**ystems, 1998-2001) and **EVANET-HYDRA** (**E**valuation and **NET**work of EC-Decision Support Systems in the Field of **HYDR**ological Dispersion Models and of **A**quatic Radioecological Research, 2001-2004).

1.3 Input/Output Files

Several files are used to save or load all the information related to our problem. First, there is a file used to save all the information related to our problem, e.g., the appearance and general information of the tree, component utilities, intervals and average normalized weights and alternative names and consequences. This is called **workspace** file.

Second, partial files are used to save only component utilities, alternative names and consequences or weights in the hierarchy. When we are trying to load one of these files, the module warns if the partial file does not match up with the current workspace features. For example, if we try to load an alternatives file, the module checks if the number of consequences of an alternative is equal to the number of leaves in the current workspace. This file also saves the number of intermediate nodes and the leaf names which are checked too.

Remember that a workspace file can be loaded or saved using the **WorkSpace** menu (Figure 1.5) or using the respective button in the toolbar (Figure 1.9).

Partial files can be loaded or saved using the **File** menu, see Figure 1.14.

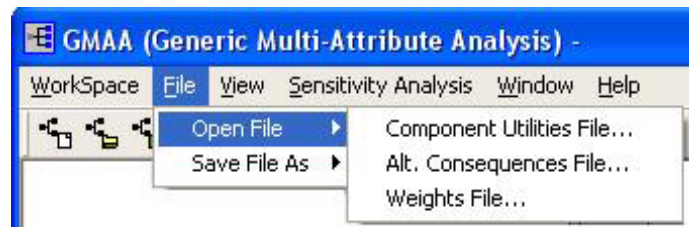


Figure 1.14. File menu

Users can define/modify all the information related to their multi-attribute decision problem, so there does not appear to be any reason why they would need to know what these files are like. However, our module may be called by other programs. For example, the input alternative consequences may be generated by another program, which must know the appearance of the alternatives files.

Now let us look at the appearance of the different files. All files begin with a key word used to identify the kind of file to be opened or saved. If we are opening or saving an alternatives file, the module checks the key word, which should be **strategy**.

The workspace file appearance is :

```

| WorkSpace
| INTERMEDIATE_NODE NUMBER
| *num_node_int
|
| *intermediate node1 description
| .....
| *intermediate noden description
|
| LEAF NODES
| *num_leaves
| STRATEGY NUMBER
| *num_strategies
|
| *leaf_node1 description
| .....
| *leaf_noden description
|
| STRATEGY NAMES
| *strat_name1
| .....
| *strat_namen

```

where the *intermediate_node_i description* has the appearance:

```

| *node_name
| *node_label
| *father_name
| *node_description
| *pos_x * pos_y
| *height * weight
| *num_sons *tradeoffs
| *son_name *min_weight *nor_weight *max_weight *tradeoff_low *tradeoff_upp
| .....
| *son_name *min_weight *nor_weight *max_weight *tradeoff_low *tradeoff_upp

```

where **tradeoffs* points out if the method based on trade-offs was used to elicit the weights for the subobjectives stemming from it, and **tradeoff_low* and **tradeoff_upp* are the lower and the upper end-points of the probability intervals provided by the DM in the weight elicitation methods.

The *leaf_node_i description* consists of:

```
| *node_name
| *node_label
| *node_units
| *recmin *recmax
| *father_name
| *node_description
| *pos_x * pos_y
| *height * weight
| val/sc *min_sc *max_sc *min_val *max_val
| .....
| val/sc *min_sc *max_sc *min_val *max_val
|
| [0,1,2] //0 a subjective scale is being used for this attribute
|           //1 a component utility function is defined in the next three lines
|           //2 imprecise utilities are assigned for discrete values
|
| *Mininum_utility_function
| *Average_utility_function
| *Maximum_utility_function
```

where the **Mininum_utility_function*, **Average_utility_function* and **Maximum_utility_function* define the class of utility functions for this attribute and have the following appearance:

**cubic_spline *bounds *characteristics*

“cubic_spline” is the cubic spline coefficients

$(a_1, b_1, c_1, d_1; a_2, b_2, c_2, d_2; a_3, b_3, c_3, d_3; a_4, b_4, c_4, d_4)$

that define the utility function in four intervals

interval1 $\rightarrow a_1 + b_1x + c_1x^2 + d_1x^3$

interval2 $\rightarrow a_2 + b_2x + c_2x^2 + d_2x^3$

$$\text{interval3} \rightarrow a_3 + b_3x + c_3x^2 + d_3x^3$$

$$\text{interval4} \rightarrow a_4 + b_4x + c_4x^2 + d_4x^3$$

Five numbers form “bounds” defining the intervals where cubic splines are applied:

bound1 bound2 bound3 bound4 bound5

so that

[*bound1*, *bound2*] is the first interval,

.....

[*bound4*, *bound5*] is the fourth interval,

Obviously, *bound1* and *bound5* represent the minimum and maximum attribute range.

Finally, “characteristics” includes the minimum and maximum attribute range, the worst and the best values, a number that represents whether the function is increasing or decreasing (1→ decreasing, 0→ increasing) and a number that whether it is a piecewise linear utility function or not.

In the case of imprecise utilities for different discrete values, instead of using

| **Minimum_utility_function*

| **Average_utility_function*

| **Maximum_utility_function*

the following information must be provided:

| **number_of_values*

| **value₁ *util_min₁ *util_avg₁ *util_max₁ *descr₁*

| ...

| **value_n *util_min_n *util_avg_n *util_max_n *descr_n*

In the case of subjective scale we will just enter

| 0

Note that in lines

| **val/sc** **min_sc *max_sc *min_val *max_val*

|

| **val/sc** **min_sc *max_sc *min_val *max_val*

the string **sc** points out that a subjective scale is being used. As a consequence, **min_val* = **max_val* = 0.0. The string **val** points out that a component utilities have been identified for that attribute, and **min_sc* = **max_sc* = 0.0.

The weights file appearance is as follows:

```
| weight
|
| Num_Leaves: *num_hoj
| Num_Intermediate_nodes: *num_int
|
| *leaf_name1
| .....
| *leaf_namen
|
| *node_name *num_sons *trade - offs
|   *w_min1 *w_avg1 *w_max1 *w_trademin 1 *w_trademax 1
|   .....
|   *w_minn *w_avgn *w_maxn *w_trademin n *w_trademax n
| .....
| *node_name *num_sons *trade - offs
|   *w_min1 *w_avg1 *w_max1 *w_trademin 1 *w_trademax 1
|   .....
|   *w_minn *w_avgn *w_maxn *w_trademin n *w_trademax n
```

where $*w_avg_i$ is the average normalized weight of the i -th objective, and $*w_min_i$ and $*w_max_i$ are the lower and the upper end-points of the normalized weight intervals. When $*trade - offs$ is equal to 1, the method based on trade-offs was used and $*w_trade_{min i}$ $*w_trade_{max i}$ are the DM's answer to the i -th probability question. Otherwise, a direct assignment was used and $*w_trade_{min i}$ $*w_trade_{max i}$ are the weight intervals directly provided by the DM.

The component utilities file appearance is as follows:

```
| utility
|
| Num_Leaves: *num_hoj
|
| *leaf_name1
| .....
| *leaf_namen
```



```
|
| [0,1,2] //0 means that a component utility function is defined in the next three
lines
```

```
| //1 means that imprecise utilities are assigned for discrete values
```

```
| //2 means that subjective values are being used in this attribute
```

```
| utility_function_1
```

```
| ...
```

```
| [0,1,2]
```

```
| utility_function_n
```

where the *utility_functions* lines are the same as explained for the workspace files.

The alternatives file appearance is as follows:

```
| strategy
```

```
|
```

```
| Num_Leaves: *num_leaves
```

```
| Num_Strat: *num_strategies
```

```
|
```

```
| *leaf_name_1
```

```
| .....
```

```
| *leaf_name_n
```

```
|
```

```
| *strategy_name_1
```

```
| .....
```

```
| *strategy_name_n
```

```
|
```

```
| *leaf_name_1
```

```
| val/sc *min_sc_1 *max_sc_1 *min_val_1 *max_val_1
```

```
| .....
```

```
| val/sc *min_sc_m *max_sc_m *min_val_m *max_val_m
```

```
|
```

```
| .....
```

```
|
```

```
| *leaf_name_n
```

```
| val/sc *min_sc_1 *max_sc_1 *min_val_1 *max_val_1
```

|

| **val/sc** *min_sc_m *max_sc_m *min_val_m *max_val_m

where the string **sc** points out that a subjective scale is being used. As a consequence, *min_val_i = *max_val_i = 0.0. The string **val** points out that a component utilities have been identified for that attribute, and *min_sc_i = *max_sc_i = 0.0.

For more information or doubts on the input/output file, get in touch with the authors.

2 BUILDING AN OBJECTIVES HIERARCHY

There are several benefits to be gained from using a hierarchy to model complex decision-making problems with multiple objectives. For instance, it helps to ensure that there will be no big gaps (missing objectives) at lower levels, situations where redundancy or double-counting could easily occur can be identified and it provides a basis upon which to develop and appraise screening criteria (Brownlow and Watson, 1987). It is important to remember that the consequences associated with the alternatives are measured in terms of the attributes in the leaves of the tree.

The DM can create or delete nodes and branches to build or modify the objectives hierarchy. A floating menu is displayed when the DM left-clicks a node of the tree (Figure 2.1). This floating menu is composed of the node name and two options: **Create a son** and **Delete a Branch**. If we select the **Create a son** option, the node becomes an intermediate node and a new leaf is created. The default weights associated with nodes stemming from the upper-level objective will be equal, and the sum of these weights will, of course, be 1.

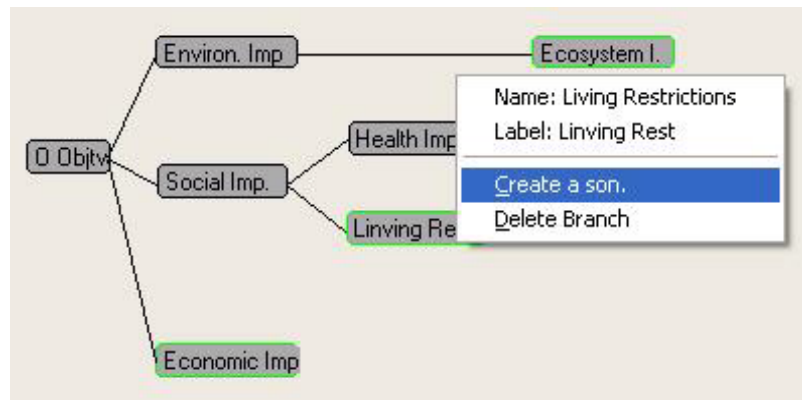


Figure 2.1. Floating menu to change the appearance of the hierarchy

As we can see, the lowest-level objective edges are green, while the remaining are black. Once the DM has specified preferences and the alternative consequences have been entered in terms of the attribute associated with the lowest-level objectives, their colour turns to blue.

The new node leaf label and name are assigned randomly, and there is no description or units. This information can be viewed in the **Node Information** window, which is displayed by right-clicking a node of the tree, see Figure 2.2. All the leaf related

information can be changed directly by overwriting the default values. The leaf name, label and units must be less than 25, 13 and 49 characters long, respectively. If the **Delete Branch** option is chosen in the floating menu, see Figure 2.1, this node and all nodes stemming from it will be removed.

Once the DM has built the objectives hierarchy, the attribute units and range must be entered in the leaves. Again, default values are suggested, which the DM can rewrite, see Figure 2.3.

Note that if the DM modifies the range, he/she will have to enter new component utilities and alternative consequences for the corresponding attribute.

The DM must also point out which attributes have a subjective scale by selecting the **Subjective Scale** folder in the respective **Node Information** window and then clicking the **Using Subjective Scale** check button, see Figure 2.4.

This window will also be used in the next section to enter/modify the respective subjective attribute values.

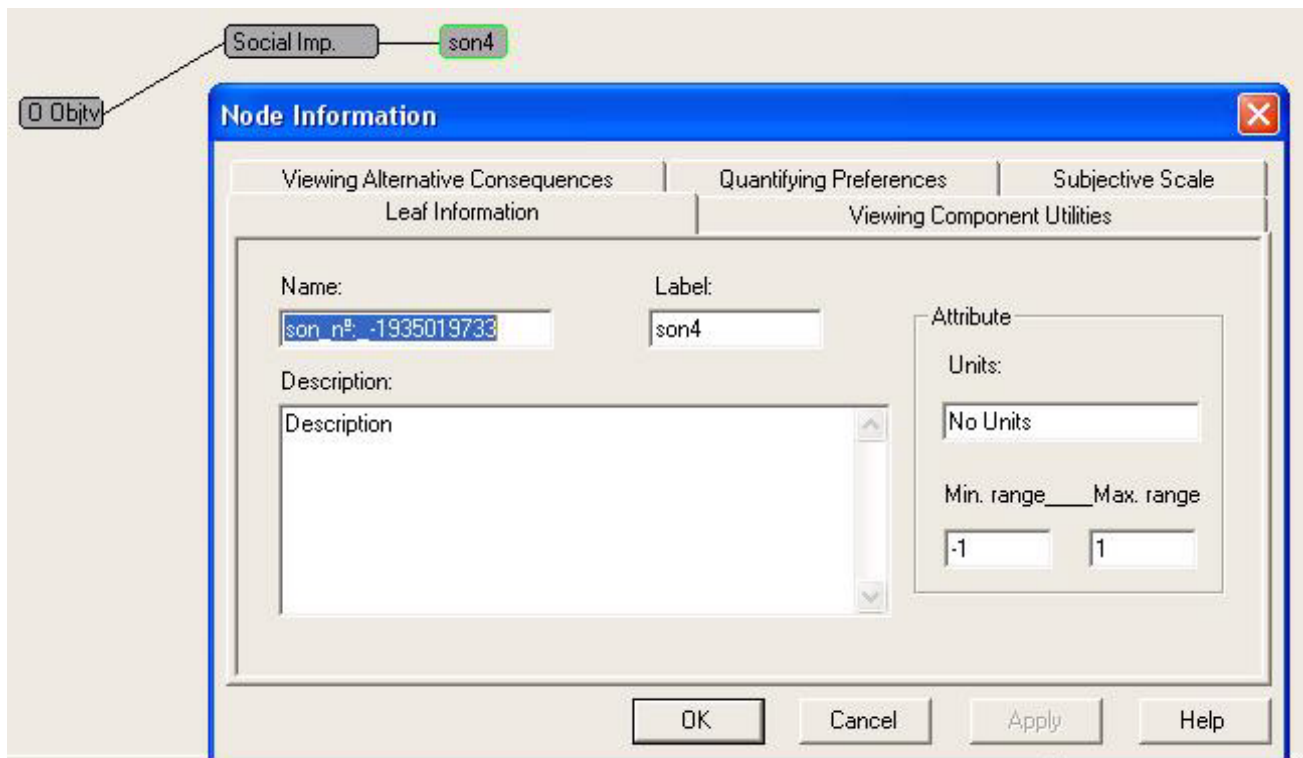


Figure 2.2. Creating a new node

The 'Node Information' dialog box has a blue title bar with a close button. It contains several tabs: 'Quantifying Preferences', 'Subjective Scale', 'Weight Stability Interval', 'Leaf Information', 'Viewing Component Utilities', and 'Viewing Alternative Consequences'. The 'Quantifying Preferences' tab is active. It features a 'Name' field with 'Dose to Critical Indiv.', a 'Label' field with 'D. Cr. Indiv.', and a 'Description' text area containing: 'The effective dose received by individuals belonging to a critical group living in the area, drinking water, and eating aquatic food and terrestrial food irrigated with water from the contaminated water body. Evaluated by MOIRA dose model.' To the right, the 'Attribute' section shows 'Units' as 'Effective Dose in milliSieve' and 'Min. range' as '0.76' and 'Max. range' as '2.47'. At the bottom are 'OK', 'Cancel', 'Apply', and 'Help' buttons.

Figure 2.3. Entering the attribute units and range

The 'Node Information' dialog box is shown with the 'Subjective Scale' tab active. It displays a grid of 10 vertical sliders, each with a '1 -' at the top and a '0 -' at the bottom, all set to '0.00'. Below the sliders are 10 text boxes containing: 'no actio', 'fish ban:', 'fish ban:', 'lake limi', 'l.liming +', 'potash t', 'potash t', 'fertilizati', 'fertiliz +', and an empty box. At the bottom, there is a '<<Back' button, radio buttons for 'min' (selected) and 'max', a checked checkbox for 'Using Subjective Scales', and a 'Next>>' button. The 'OK', 'Cancel', 'Apply', and 'Help' buttons are at the very bottom.

Figure 2.4. Indicating that an attribute has a subjective scale

3 ALTERNATIVES PROCESSING

Before entering the alternative consequences, attribute units and ranges have to be entered in the leaves of the objectives hierarchy, because the system will check if the entered values are consistent with the attribute ranges. Also, attributes using a subjective scale must be indicated.

Alternative names and consequences can be loaded from file by selecting a **Workspace file** or an **Alt. Consequences file**. The appearance of the workspace and alternative files was described in section 1.2. In the second case, the system checks the consistency between the alternative file information and the current workspace. Leaf number and names must match up. The system also reports alternative consequences that are out of range to the DM.

These consequences can be viewed by selecting the **View Alt. Consequences** option in the **View** menu or clicking the respective toolbar button (the window displayed is shown in Figure 3.1) or by left-clicking the respective leaf of the hierarchy and selecting the **Viewing Alternative Consequences** folder, Figure 3.2.



	No Actions	Fish Bans (1st)	Fish Bans (2,3,4)	Lake Limin.
Ecosystem I.	5.000	4.500	4.250	2.000
Cost t Image	0.000	0.000	0.000	0.700
D. Cr. Indiv	2.335	2.050	1.050	2.300
Dur. of Bans	3.000	5.000	33.000	3.000
Cost Applica	0.000	12.500	30.000	160.000
Collective D	68.150	60.500	24.000	65.500
Cost Economy	0.000	150.000	403.000	20.000

Figure 3.1. Viewing the alternative consequences

Different icons are used to remind us whether subjective scales, utilities for discrete attribute values or a utility function are being used for one or more leaves. All of these are used to quantify the DM's preferences concerning the alternative consequences and

will be explained in the next section. In this example, all the attributes, except *Cost to Image*, use utility functions, see Figure 3.1. A subjective scale is defined in the *Cost to Image* leaf.

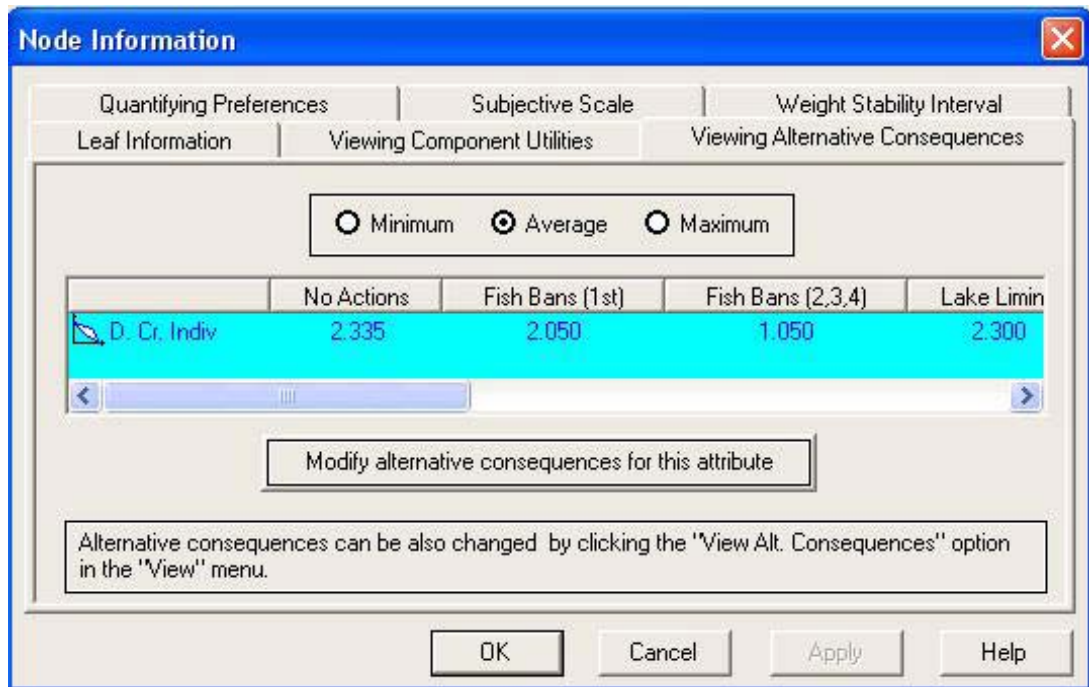
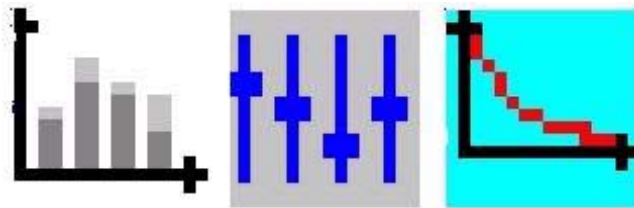


Figure 3.2. Alternative consequences for a specific leaf



Icons representing utilities for discrete attribute values, subjective values and a utility function, respectively

Looking at the window displayed in Figure 3.1, we realize that we have the option of adding alternatives, modifying alternative consequences and deleting alternatives. If we want to create a new alternative manually we just have to click the **Add Alternative** button and the window below will be displayed:

Alternative Name:

Enter for the different attributes the respective imprecise consequence.

Attribute	min	max	[recmin , recmax]
Ecosystem Index	0.0	0.0	[1.000 , 5.000]
Cost to Image	0.0	0.0	Subjective scale [0,1].
Dose to Critical Indiv.	0.0	0.0	[0.760 , 2.470]
Duration of bans	0.0	0.0	[0.000 , 36.000]
Cost of Application	0.0	0.0	[0.000 , 702.000]
Collective Dose	0.0	0.0	[20.300 , 72.300]
Cost to Economy	0.0	0.0	[0.000 , 426.000]
	0.0	0.0	

<< Back Next >>

Cancel OK

Figure 3.3. Insert a new alternative

The DM must provide an alternative name (less than 25 characters long) and consequences for all the attributes in the objectives hierarchy taking into account their features. The system reminds us of attributes using a discrete values set, attributes using a subjective scale and attributes for which a linear/non-linear utility function has been identified. In the first case, only permitted attribute values can be entered. Note that the system also reminds us what the permitted attribute are. When using a subjective scale, the values entered must be greater than or equal to 0 and less than or equal to 1. Finally, when a utility function has been identified, entered values must be within the respective attribute range. The **Next>>** and **<<Back** buttons must be used when more than eight leaves are defined in the objectives hierarchy.

Moreover, the system accounts for uncertainty about the alternative consequences by means of uniformly distributed attribute value intervals in the first two cases. Note that the situation under precision or under certainty will be the particular case in which the extremes of each interval are the same. Remember that the DM does not have to provide the average alternative consequences, they are assessed by the system as in the case of modifying strategy values.

Once we have entered the new alternative consequences and name and clicked the

OK button, the system checks the consistency of the values entered and reports alternative consequences that are out-of-range or illegal discrete attribute values, see Figure 3.4.

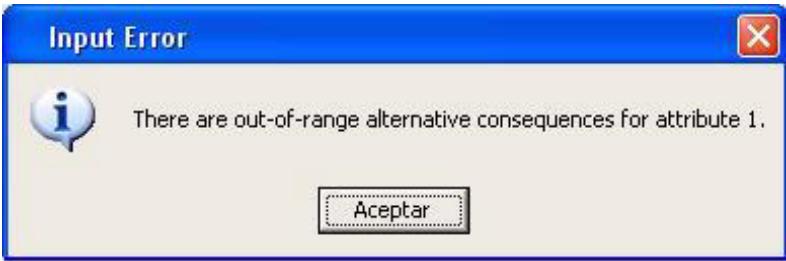


Figure 3.3. Insert a new alternative

The DM can save the alternative consequences to file using the **Save Workspace** or **Save Workspace As** options in the **Workspace** menu, or the **Alt. Consequences File** option in the **Save File As** submenu of the **File** menu.

Note that subjective values can also be entered or modified directly using the **Subjective Scale** folder in the respective **Leaf Information** window, see Figure 3.5.

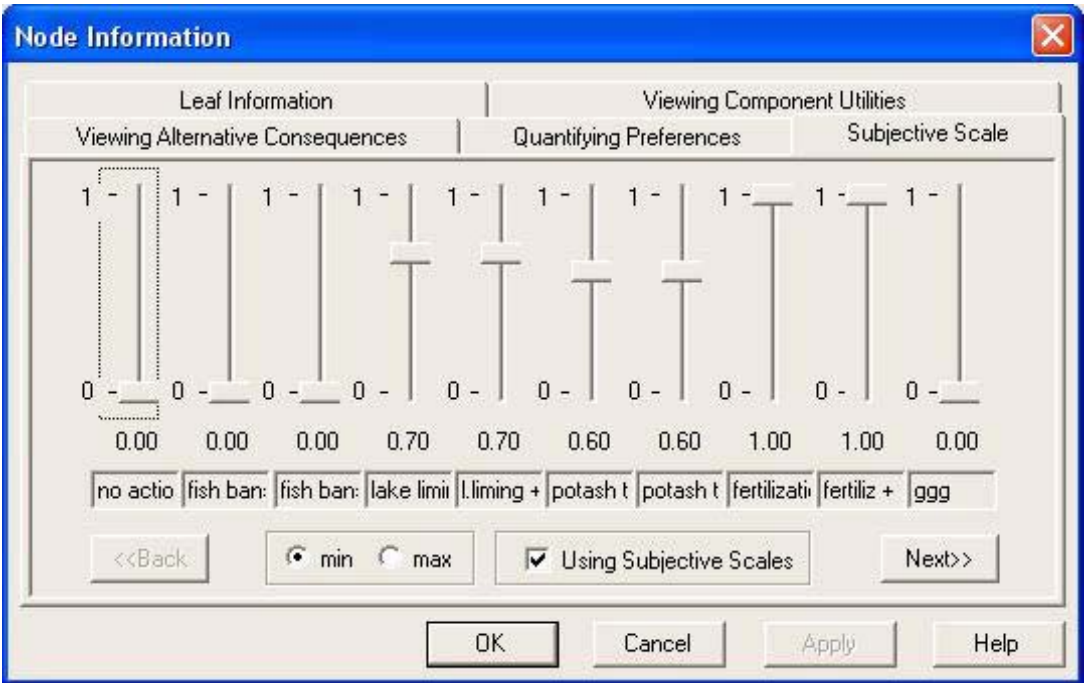


Figure 3.5. Subjective Values

Again, instead of entering only one (precise) attribute value for each alternative, DMs can use the scrollbars to provide a range of responses, which is less stressful. Minimum and maximum values will be entered depending on the active radio button. The **Next>>** and **<<Back** buttons must be used when more than ten alternatives have been identified.

To remove an alternative, the DM should select the alternative name from the drop list on the left-hand side of the **Modify Consequ.** button, see Figure 3.6. This activates **Delete Alternative** and **Modify Consequ.**, and the DM can delete the alternative by clicking the **Delete Alternative** button.



Figure 3.6. Deleting an existing alternative (*Fish Bans (1st)*)

Finally, to modify alternative consequences or the alternative name, the DM should click the **Modify Consequ.** button instead of the **Delete Alternative** after selecting the alternative name from the drop list and the window shown in Figure 3.7 will be displayed.

This window displays the current values for the alternative whose consequences and/or name the DM has chosen to modify. To change the alternative name, the DM must overwrite it in the respective label. To modify the alternative consequences, the DM must double-click on the leaf names, and a window in which the values can be modified will be displayed, Figure 3.8.

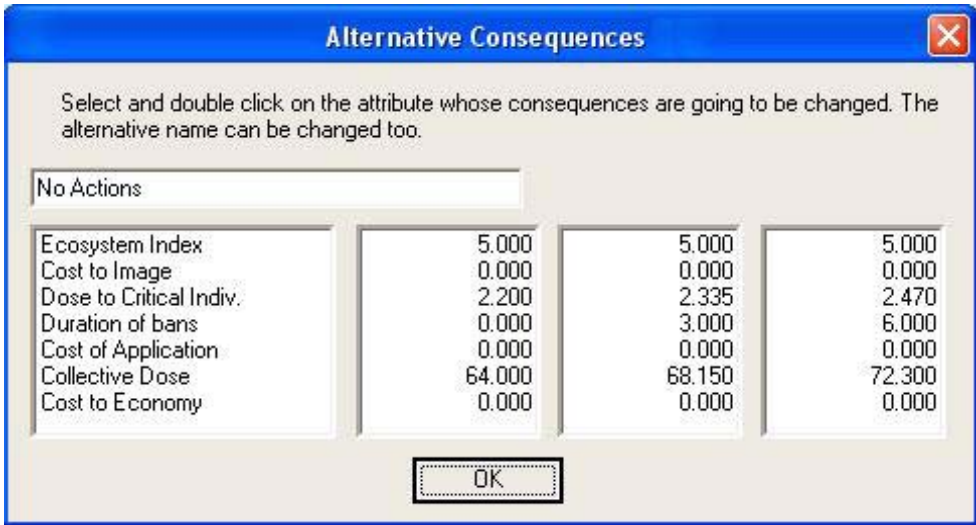


Figure 3.7. The current consequences for a specific alternative *No Actions*

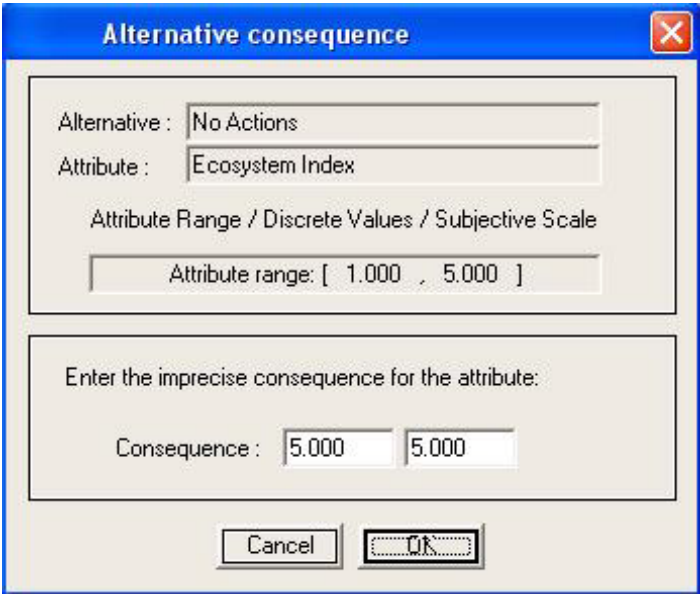


Figure 3.8. Changing consequence values

The system reminds us of the attribute range when a utility function has been identified for the attribute, the permitted discrete attribute values or whether a subjective scale is being used. Current default values are also presented, which the DM can change. The system checks the consistency of the values entered and reports out-of-range alternative consequences or illegal discrete attribute values, see Figure 3.9.

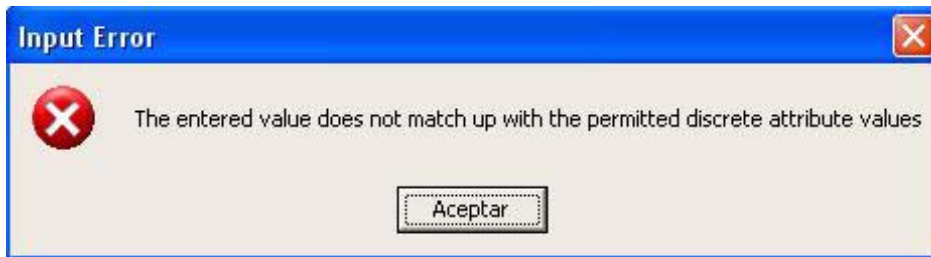


Figure 3.9. Input error message for attributes with discrete values

Finally, we should point out that although we have stated previously that alternative consequences are entered in the system prior to the quantification of the DM's preferences, for consistency reasons, especially as regards attributes with discrete values, the DM is asked to quantify his/her preferences for the different attributes before entering the alternative consequences, see Figure 3.10.



Figure 3.10. Reminder to quantify preferences

The alternative consequences for an attribute can be also modified by left-clicking the respective leaf of the hierarchy, selecting the **Viewing Alternative Consequences** folder and clicking the **Modify alternative consequences for this attribute** button, see Figure 3.2. The window shown in Figure 3.11 will be displayed, in which the DM is reminded the attribute name, range and units, and the alternative names and their current imprecise consequences.

The **Next>>** and **<<Back** buttons must be used when more than eight alternatives have been identified. The alternative consequences can be modified by rewritten them.

Note that we are not allowed to modify subjective values in this window. The system will report this situation to us, see Figure 3.12.

Modifying alternative consequences for an attribute

Attribute Name:

Attribute Range:

Attribute Units:

Enter for the different alternatives the respective imprecise consequence:

Alternative names	min	max
No Actions	<input type="text" value="5.000"/>	<input type="text" value="5.000"/>
Fish Bans (1st)	<input type="text" value="4.000"/>	<input type="text" value="5.000"/>
Fish Bans (2,3,4)	<input type="text" value="3.500"/>	<input type="text" value="5.000"/>
Lake Liming	<input type="text" value="1.500"/>	<input type="text" value="2.500"/>
L.Liming + Fish Bans (3)	<input type="text" value="1.500"/>	<input type="text" value="2.600"/>
Potash Treatment	<input type="text" value="1.800"/>	<input type="text" value="2.700"/>
Potash T. + Fish Bans (3)	<input type="text" value="1.800"/>	<input type="text" value="2.600"/>
Fertilization	<input type="text" value="1.000"/>	<input type="text" value="1.600"/>

<< Back Next >>

Cancel OK

Figure 3.11. Modifying alternative consequences in an attribute

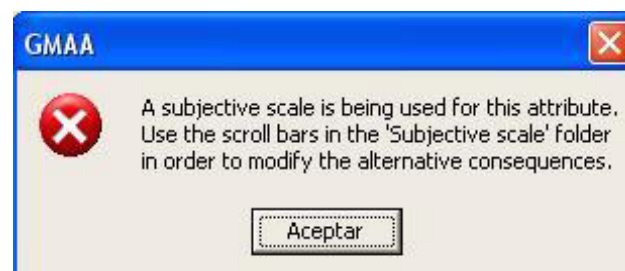


Figure 3.12. Reminder to use the "subjective scale" folder

4 QUANTIFYING PREFERENCES

Quantifying preferences involves assessing the DM's component utilities, which represent the DM's preferences concerning the possible alternative consequences in the respective attributes, and the relative importance of criteria. Both will be used later to evaluate alternatives through the multi-attribute utility function.

In both cases, the system admits incomplete information through value intervals as responses to the probability questions the DM is asked, which leads to classes of utility functions and weight intervals, respectively. This is less stressful on experts, see Weber (1987), von Nitzsch and Weber (1998) and Ríos et al. (1994). Moreover, this makes the system suitable for group decision support, because individual conflicting views or judgements in a group of stakeholders can be captured through imprecise responses.

4.1 Assessment of component utilities

The user has two alternative ways to view the component utilities: either by selecting the **Viewing Component Utilities** folder in the respective leaf of the tree, as shown in Figure 4.1, or through the main menu, by selecting the **View Component Utilities** option in the **View** menu (Figure 4.2a and Figure 4.2b).

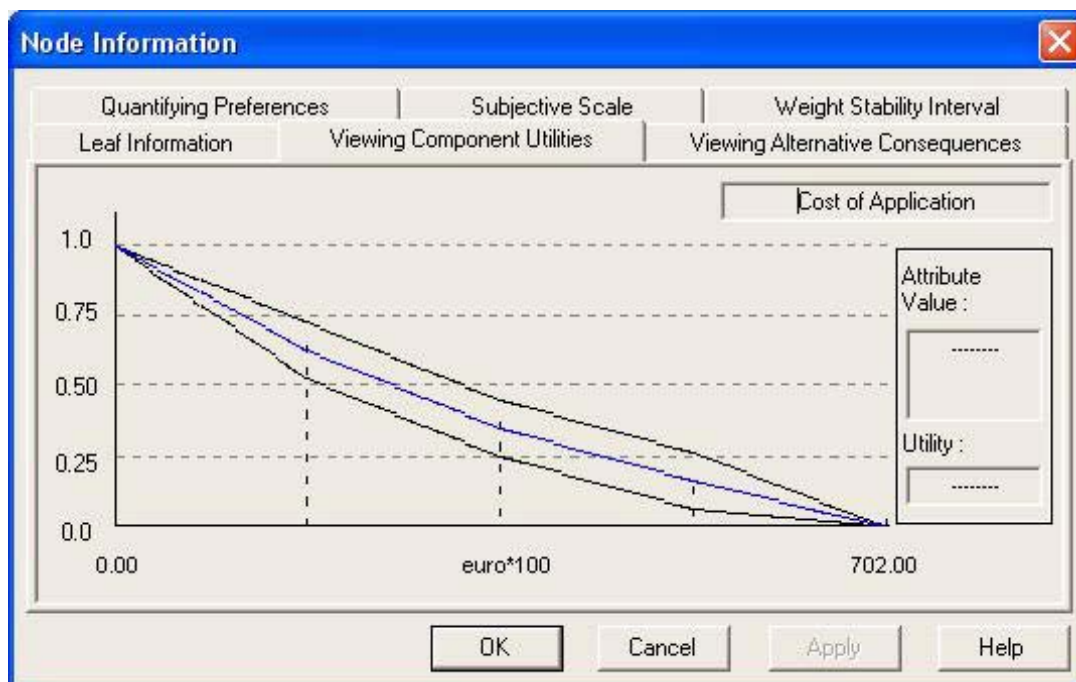


Figure 4.1. Viewing component utilities

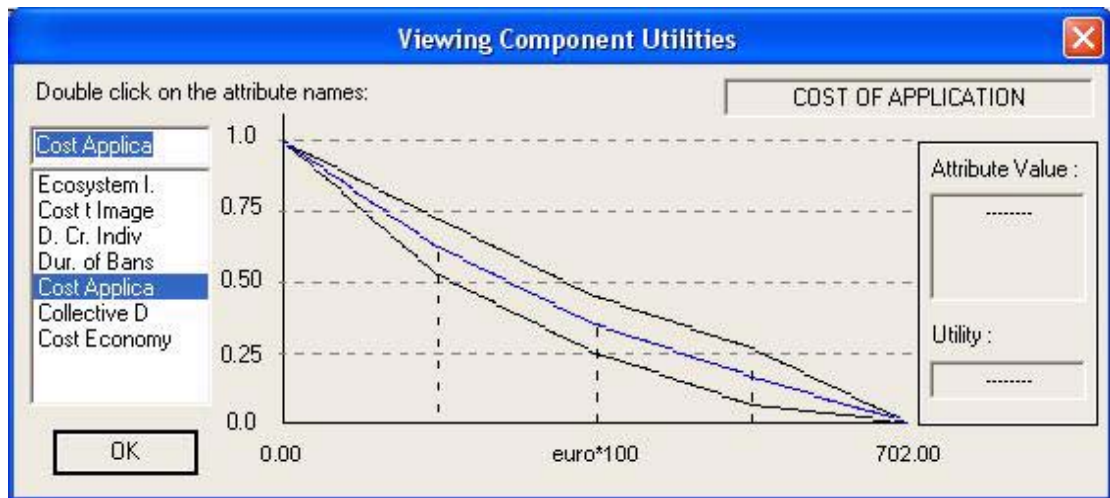


Figure 4.2a. Viewing component utilities

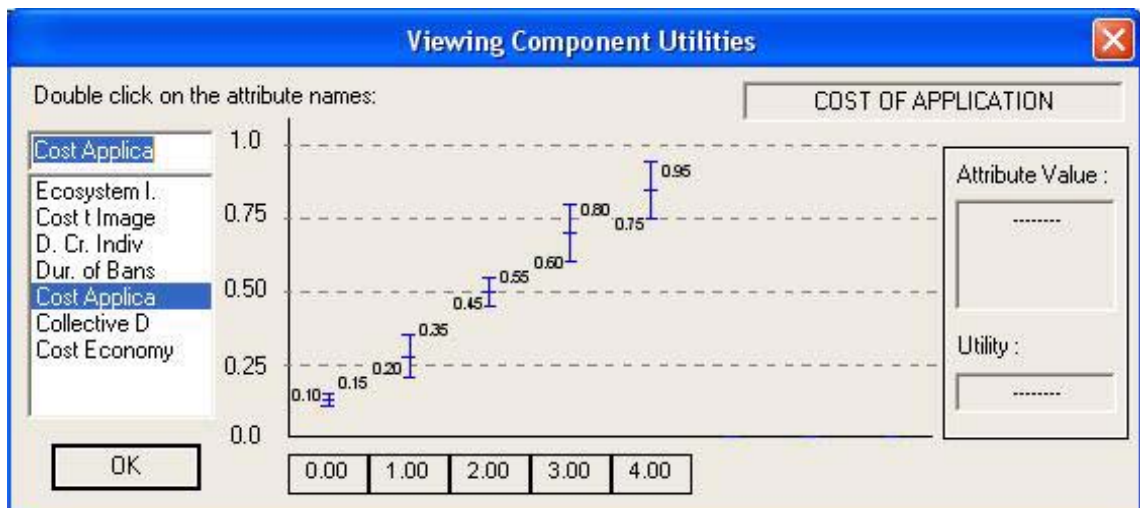


Figure 4.2b. Viewing component utilities

The DM can change any component utility with whose appearance he/she does not agree. Users can choose between constructing an imprecise piecewise linear utility function (providing up to three intermediate attribute values and their respective imprecise utilities or using a method based on the combination of two slightly modified standard procedures for utility assessment, the *Fractile Method* (CE-Method) and the *Extreme Gambles Method* (PE-Method), based on imprecise assignments by means of intervals) or providing utilities for discrete attribute values.

Component utilities can be defined/modified by selecting the **Quantifying Preferences** folder in the respective leaf. Here, the DM can choose between the methods provided by the system by selecting the respective radio button and then, clicking the **Next>>** button, see Figure 4.3.

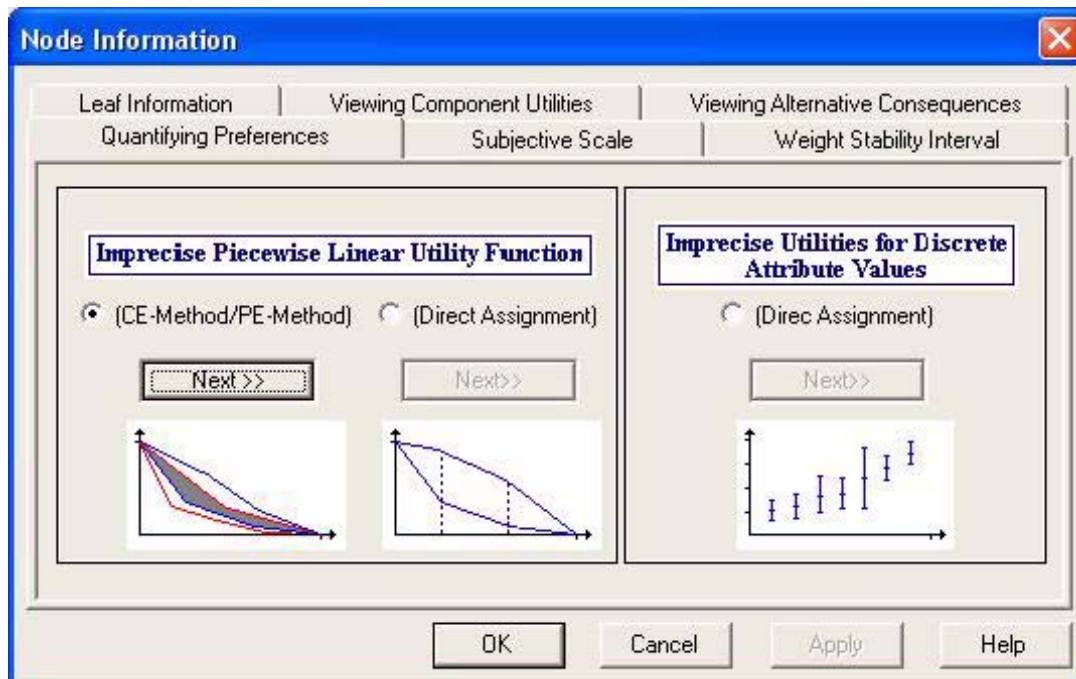


Figure 4.3. Selecting a method to modify/define component utilities

4.1.1 Imprecise Piecewise Linear Utility Functions (Direct Assignment)

When there is a deep and precise knowledge about the attribute, the DM can directly construct an imprecise piecewise linear utility function, and the window shown in Figure 4.4 is displayed. In this **General attribute features** window, the DM can change/enter the units, minimum and maximum range and the shape (monotonically increasing/ monotonically decreasing) of the piecewise linear utility function. Note that if a piecewise linear utility function was already being used in this attribute, the user is reminded the values provided above.

Once the user agrees with the values and has clicked the **Next>>** button, a new window is displayed (Figure 4.5).

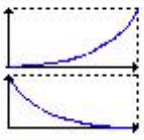
General attribute features

Specify the minimum and maximum ranges for the attribute :

Minimum Range: Maximum Range:

☐ Monotonically Increasing

☒ Monotonically Decreasing



Units :

<<Back

Next>>

Figure 4.4. General attribute features

Linear Utility Function

The lower and upper bounds can be changed: (1 , 5)

Enter up to three intermediate points for the attribute so as Point1.x < Point2.x < Point3.x :

☒ Point 1: Attribute Value (x) : 1.1 Utility: (0.85 ; 0.95)

☒ Point 2: Attribute Value (x) : 1.5 Utility: (0.45 ; 0.65)

☒ Point 3: Attribute Value (x) : 2 Utility: (0.1 ; 0.3)

View

<<Back

Save

OK




Figure 4.5. Piecewise Linear Utility Function

The user is asked to provide intermediate attribute values (up to three). Instead of demanding only one (precise) utility for each attribute value, DMs are allowed to

provide a utility interval, which is less stressful on experts, and both endpoints being equal would be equivalent to the precise case. The imprecise utility function bounds will be built by joining up to five linear segments between the best and worst attribute values.

If no intermediate points are specified, then the result will be a single linear function. The constructed linear function will not be viewed (at the bottom of the window) until the **View** button is clicked. Utility function ranges can be redefined again in this window.

If dissatisfied with the shape of the new piecewise linear function, the user can change the ranges and intermediate points entered by rewriting them in the respective boxes and click the **View** button again to view the changes. The shape of the utility function can also be changed by moving the active points with the mouse, corresponding to the intermediate points entered by the user. When the cursor is on an active point it changes from an arrow to a cross, then, if the user left-clicks on it, the active point can be dragged by the mouse. Note that the system does not allow the user to introduce inconsistent values. For instance, if the utility function is increasing the upper utility corresponding to the second intermediate point must be less than the one corresponding to the third. The <<**Back** option can also be used to go back to the previous window and modify values.

The output imprecise piecewise linear utility function can now be saved, for use for assessing the overall utility and the ranking of alternatives, but will still not be saved in the **Workspace File** or in a **Utility Function File**. The user will be warned about this.

Note that alternative consequences may become out of range due to the modification of component utilities. The system will report this situation to the DM when trying to display the alternative ranking or performing sensitivity analysis.

4.1.2 Imprecise Piecewise Linear Utility Functions (CE-Method/PE-Method)

Introduction This method for determining the imprecise piecewise linear utility function is used when the DM has little knowledge about or experience with the topic and is based on the combination of two slightly modified standard procedures for utility assessment. Several authors (see, e.g., Hershey *et al.* 1982, Jaffray 1989 or McCord

and de Neufville 1986) have suggested that, generally, elicited value/utility functions are method-dependent, and bias and inconsistencies may be generated in the elicitation process.

To overcome these problems, we use two methods jointly: the fractile method, which belongs to the class of certainty equivalent methods (CE-Method), and the extreme gambles method, included in the probability equivalent methods (PE-Method) (Farquhar 1984) (by implementing these procedures, we have provided a utility function for each attribute instead of a value one. Remember, however, that all utility functions are value functions, but not conversely). Moreover, instead of demanding only one (precise) number for each probability question, as these methods require, DMs are allowed to provide a range of responses.

As explained below, the module uses graphical representations (wheel-based colored fortune) of the assessed utility ranges to test consistency. It suggests possible inconsistencies and possible adjustments for the values that need to be reelicited by the DM.

As a result, we get a class of utility functions, rather than a single function, for each method. The responses given in both methods are compared to detect inconsistencies. There will be inconsistencies if the intersection area obtained from the two response types is empty in any range of the attribute, in which case the preferences should be reassessed.

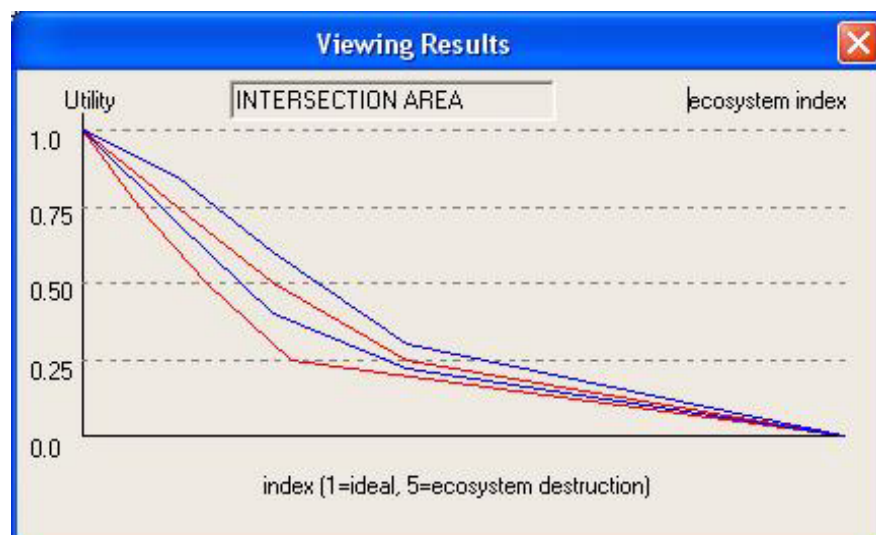


Figure 4.6. Intersection between the PE-Method and the CE-Method

These reassessments finish as soon as the DM provides a consistent range for the utility function and the intersection will represent the range for the DM's utility functions, i.e., the elicited value or utility intervals, see Figures 4.6 and 4.7. These functions were obtained from both methods: for three probability levels, $p_1 = .25$, $p_2 = .50$ and $p_3 = .75$, with the fractile method and for three selected attribute levels for the extreme gambles method.

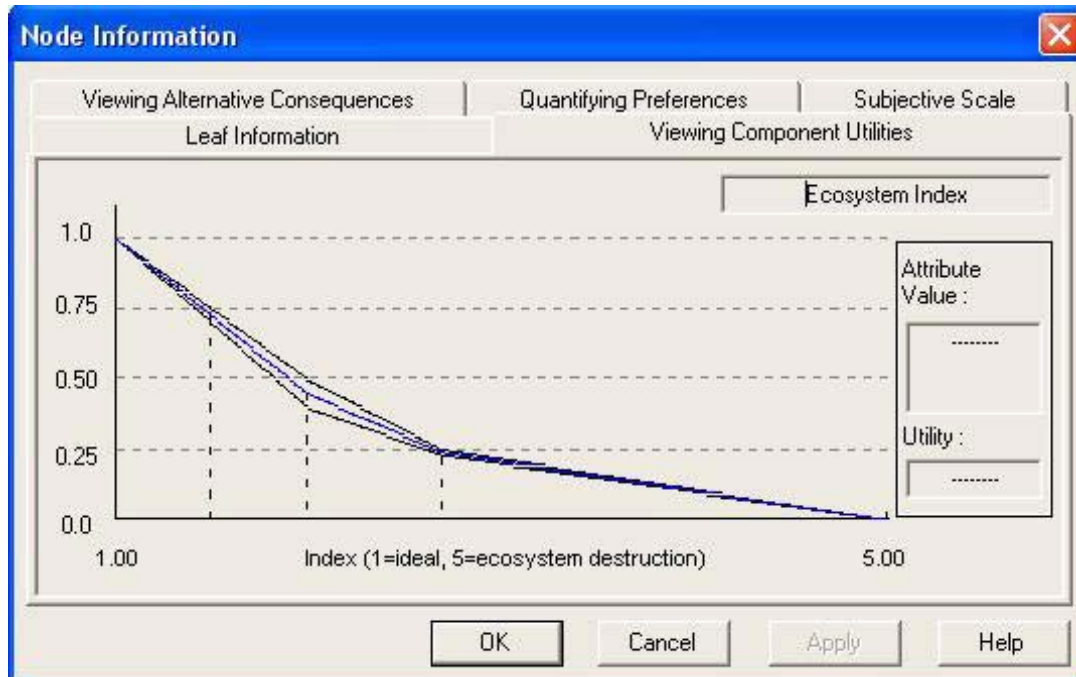


Figure 4.7. The fitted utility function

Implementation. The DM can use this method by clicking the **CE-Method/PE-Method** radio button in the window shown in Figure 4.3. The **General attribute features** window (Figure 4.4) is then displayed, as in the case of using the direct assignment.

Here the units, minimum and maximum range and shape (monotonically increasing/monotonically decreasing) of the utility function can be changed/entered. Note that default values are given if the utility function is being modified. We can choose between either keeping or modifying these values.

Once the DM has entered the values, the utility function can just be modified by clicking the **Next>>** button. Then, the first three windows referred to the (Certainty

Equivalent) CE-Method are shown in Figures 4.8, 4.9 and 4.10. The DM is asked to provide certainty equivalents for three lotteries of the type $(p, x^*; 1 - p, x_*)$, where p is equal to 0.25, 0.5 and 0.75, respectively, and x^* and x_* are the most and least preferred values for the attribute. The range given by the DM is denoted by an interval $[x_{lp}, x_{up}]$, where x_{lp} , x_{up} stand for the lower and upper bound values of a certainty equivalent for a lottery with a p -chance of yielding x^* . Figures 4.8, 4.9 and 4.10 show these lotteries.

CE-Method

Specify the sure amount interval which you consider equivalent to the lottery on the left :

(0.25 , 1 ; 0.75 , 5)

equivalent to: ([] , [])

<<Back Abort Next>>

Figure 4.8. First window of the CE-Method

CE-Method

Specify the sure amount interval which you consider equivalent to the lottery on the left :

(0.5 , 1 ; 0.5 , 5)

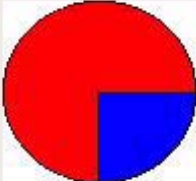
equivalent to: ([] , [])

<<Back Abort Next>>

Figure 4.9. Second window of the CE-Method

CE-Method

Specify the sure amount interval which you consider equivalent to the lottery on the left :



equivalent to: (0 , 0)

(0.75 , 1 ; 0.25 , 5)

<<Back Abort Next>>

Figure 4.10. Third window of the CE-Method

At this point, the user will be asked if he/she wants to analyze the consistency for this method. The window shown in Figure 4.11 will be displayed, where the user is asked if he/she agrees with six statements.

Consistency Checks (CE-Method)

Do you agree with the following statements ?

2.7	is preferred or equivalent to (0.5 ,	1.65	;	0.5 ,	5)	Change
2	is preferred or equivalent to (0.5 ,	1.3	;	0.5 ,	2.1)	Change
1.5	is preferred or equivalent to (0.5 ,	1	;	0.5 ,	1.65)	Change
(0.5 ,	2	;	0.5 ,	5)	is preferred or equivalent to	2.1	Change
(0.5 ,	1.5	;	0.5 ,	2.7)	is preferred or equivalent to	1.65	Change
(0.5 ,	1	;	0.5 ,	2)	is preferred or equivalent to	1.3	Change

<<Back Abort Next>>

Figure 4.11. Consistency checks for the CE-Method

If DM's previous responses were consistent, then he/she should agree with all of the statements. Otherwise, if he/she does not agree with any one of the statements, he/she should click the **Change** button at the right of the respective statement, and the window shown in Figure 4.12 will be displayed.

Changes in CE-Method

You stated that :

Statement 1: (2.1 , 2.7) was equivalent to: (0.25 , 1 ; 0.75 , 5)

Statement 2: (1.65 , 2) was equivalent to: (0.50 , 1 ; 0.50 , 5)

Statement 3: (1.3 , 1.5) was equivalent to: (0.75 , 1 ; 0.25 , 5)

You should decrease 2.7 or increase either: 1.65 or 0

OK

Figure 4.12. Making changes because the user has been inconsistent

The values that appear in the window in Figures 4.11 and 4.12 match the following DM's responses to the three probability questions for the CE-Method

$$[2.10, 2.7], \quad p = 0.25$$

$$[1.65, 2.0], \quad p = 0.50$$

$$[1.30, 1.5], \quad p = 0.75$$

In this window, the user is reminded of his/her responses to the three lotteries and is advised to modify certain values. The user can change the value directly by rewriting the new value here in this window.

Once the user agrees with the six statements and has clicked the **Next>>** button, the (Probability Equivalent) PE-Method will begin. Now, the DM has to specify three probability intervals $[p_{il}, p_{iu}]$, $i = 1, 2, 3$.

These intervals contain the indifference probabilities p of a lottery yielding x^* with probability p and x_* with a probability $1 - p$, and given sure amounts c_i . They are shown in Figures 4.13, 4.14 and 4.15.

PE-METHOD

Specify the probability interval that contains the indifference probability of a lottery yielding X_{\max} with probability p and X_{\min} with probability $1-p$, with the given sure amount:

(p , 1 ; $1-p$, 0) equivalent to : 2.7

p min. : 0 p max. : 0

<<Back Abort Next>>

Figure 4.13. First window of the PE-Method

PE-METHOD

Specify the probability interval that contains the indifference probability of a lottery yielding X_{\max} with probability p and X_{\min} with probability $1-p$, with the given sure amount:

(p , 1 ; $1-p$, 0) equivalent to : 2

p min. : 0 p max. : 0

<<Back Abort Next>>

Figure 4.14. Second window of the PE-Method

Figure 4.15. Third window of the PE-Method

As in the CE-Method, the user is now asked to analyze the consistency for this method. The consistency check is very similar to the previous one. The user must agree with the six statements shown in the window (Figure 4.16). If he/she does not agree, then he/she should click the **Change** button and modify some values.

Figure 4.16. Consistency checks for the PE-Method

The values that appear in the following window (Figure 4.17) match the answers (probability intervals) in the three lotteries for PE-Method below:

$$[0.23, 0.30] = [p_{1l}, p_{1u}]$$

$$[0.40, 0.60] = [p_{2l}, p_{2u}]$$

$$[0.70, 0.85] = [p_{3l}, p_{3u}].$$

The user can change any value by directly by rewriting the new value in this window and returning to the previous screen.

The PE-Method finishes when the user agrees with the six statements (Figure 4.16) and has clicked the **Next**>> button. Now, if there are no inconsistencies between the CE-Method and the PE-Method the user can view the results.

The can view the partial results for the CE-Method or the PE-Method and the intersection or final utility function, Figures 4.18, 4.19, 4.20, 4.21 and 4.22.

Changes in PE-Method

You stated that :

(p , ; 1-p ,) equivalent to : with p min. : p max. :

(p , ; 1-p ,) equivalent to : with p min. : p max. :

(p , ; 1-p ,) equivalent to : with p min. : p max. :

You should increase : or decrease:

Figure 4.17. Making changes due to previous inconsistent values

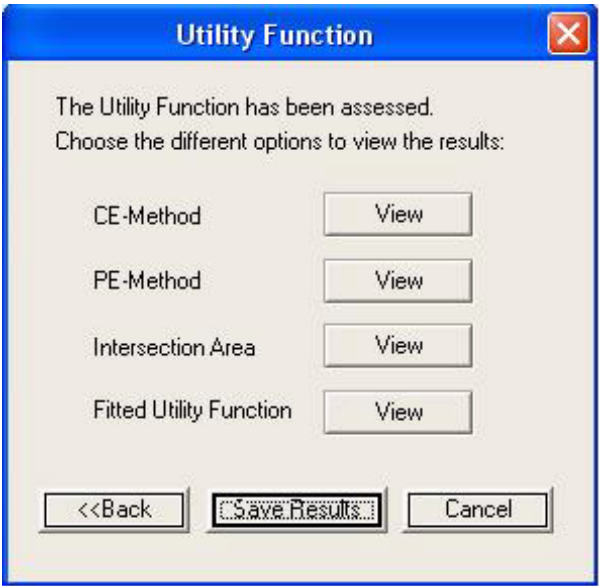


Figure 4.18. Viewing the results of utility function definition

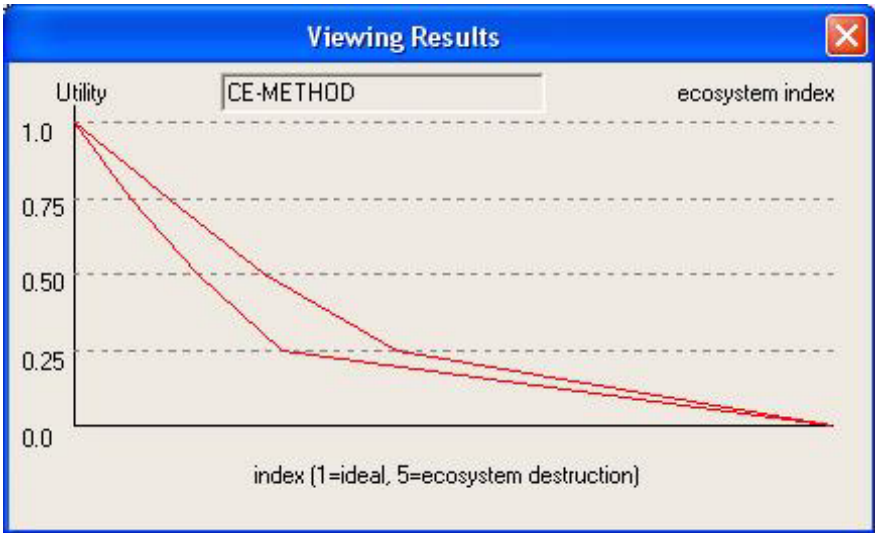


Figure 4.19. Bands limiting the utility function assessed by the CE-Method

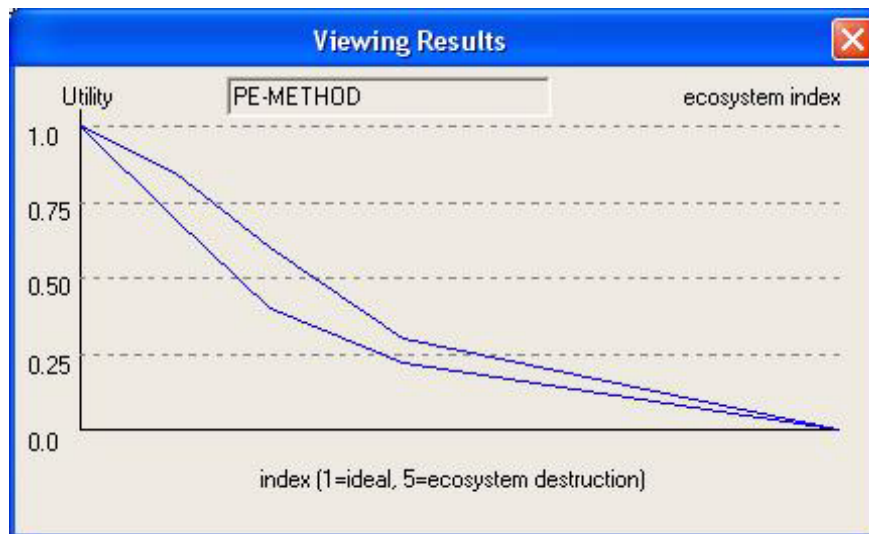


Figure 1: Figure 4.20. Bands limiting the utility function assessed by the PE-Method

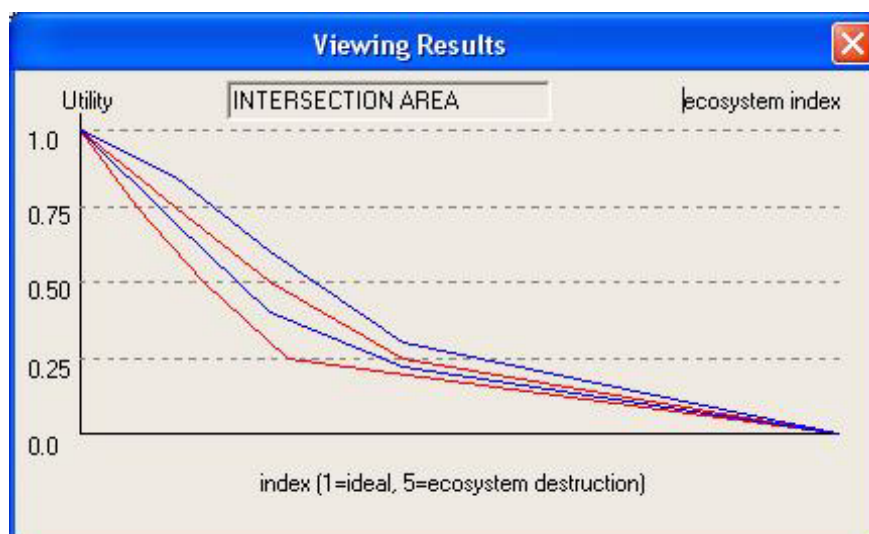


Figure 4.21. Intersection between the bands for the PE-Method and the CE-Method

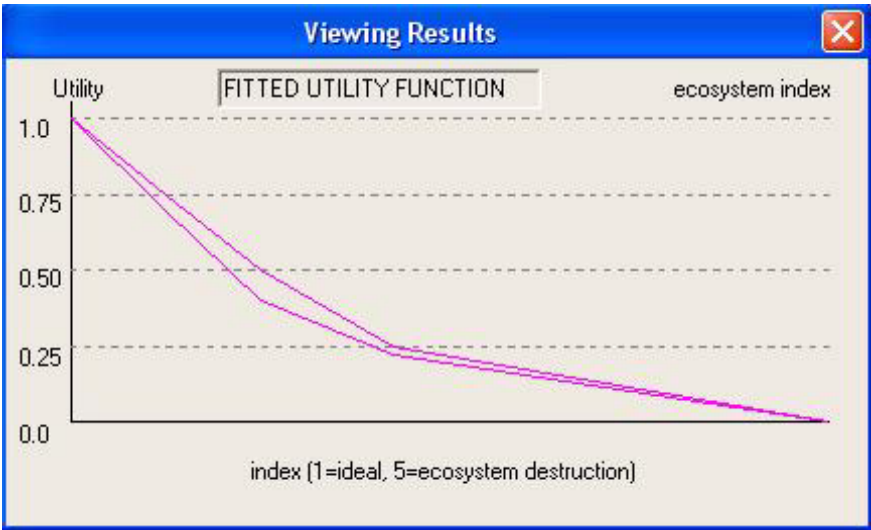


Figure 4.22a. The Fitted Utility Function

The user can now save the fitted utility function to use it for the overall values assessment just by clicking the **Save Results** button (the saved class of utility functions can be viewed in Figure 4.22b). However, he/she also can choose not to save this new utility function if he/she dislikes it.

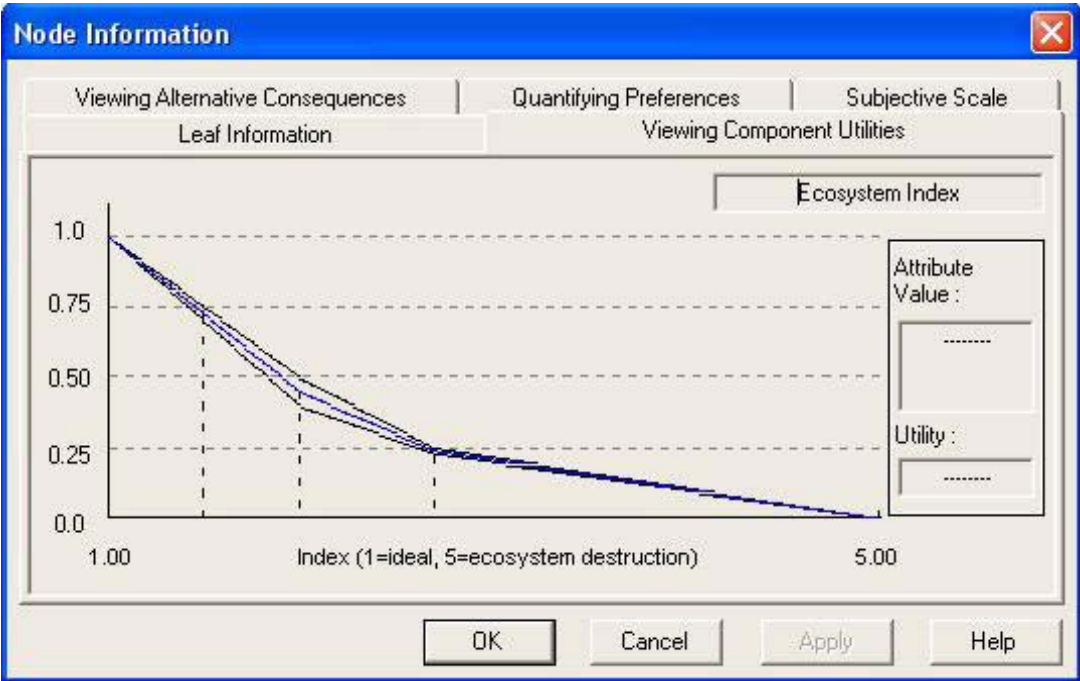


Figure 4.22b. The Fitted Utility Function

But, what happens when there is inconsistency between the CE-Method and the PE-Method? As mentioned above, results are only shown when there is no inconsistency between the two methods. The user should modify responses to achieve consistency, for which he/she will be helped by the program. With the values entered in the example shown in the above figures, there is no inconsistency.

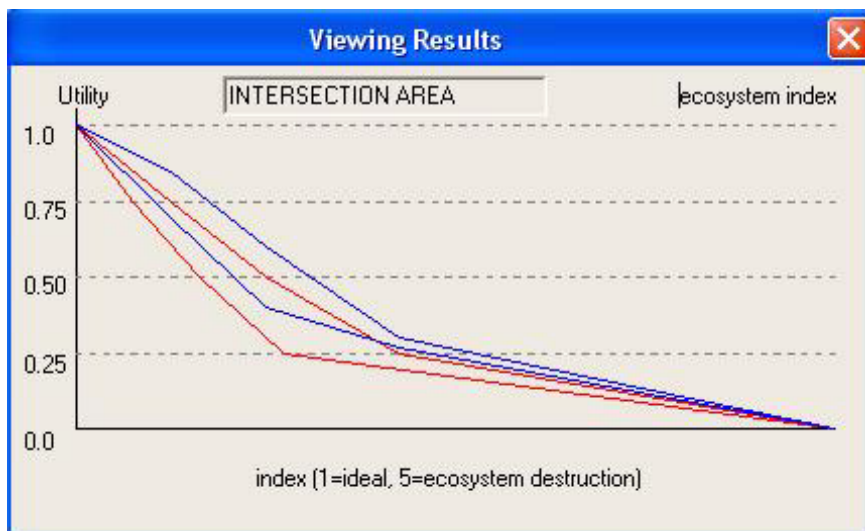


Figure 4.23. Inconsistency between the CE-Method and the PE-Method

Therefore, some changes are needed to make the results inconsistent. If the first probability interval entered by the user in the PE-Method ($[0.23, 0.30]$) is changed to $[0.27, 0.30]$, a region of the intersection between the CE-Method and the PE-Method is now empty, so there is inconsistency, as we can see in Figure 4.23.

The module will display another window (shown in Figure 4.24), in which the inconsistency is explained and we are advised to modify some values. As we can see from Figure 4.24, the user is advised to decrease the value 0.27. This value was the one that had been changed, in our case intentionally, to achieve inconsistency, so it is a logical advice.

Once this value has been changed, consistency will be achieved and the user would be able to view the results.

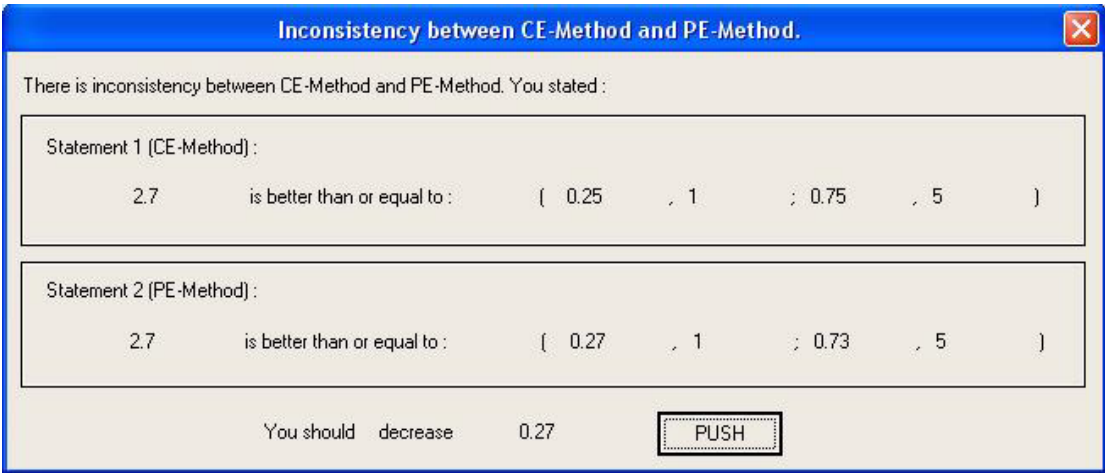


Figure 4.24. Inconsistency explanation and automatic advice

4.1.3 Imprecise Utilities for Discrete Attribute Values

In this case, imprecise utilities are assigned to different discrete attribute values. The DM can use this option by clicking the **Direct Assignment** radio button corresponding to this option in the window shown in Figure 4.3. The **Utilities for Discrete Attribute Values** window (Figure 4.25) is then displayed. Note that if discrete attribute values were already defined in this attribute, the user is reminded them.

As we can see, up to eight attribute values can be used to define the attribute utility. First, the DM has to activate the number of points he/she considers suitable through the check buttons. Then, a numeric value, its description and an imprecise utility must be provided for each one.

Utilities for Discrete Attribute Values

Enter up to eight discrete attribute values so as the attribute value and average utility for the first point is lower than for the second, and so on:

<input type="checkbox"/>	Attribute Value 1:	0	level1	Utility: (0 ; 0 ; 0)
<input type="checkbox"/>	Attribute Value 2:	0	level2	Utility: (0 ; 0 ; 0)
<input type="checkbox"/>	Attribute Value 3:	0	level3	Utility: (0 ; 0 ; 0)
<input type="checkbox"/>	Attribute Value 4:	0	level4	Utility: (0 ; 0 ; 0)
<input type="checkbox"/>	Attribute Value 5:	0	level5	Utility: (0 ; 0 ; 0)
<input type="checkbox"/>	Attribute Value 6:	0	level6	Utility: (0 ; 0 ; 0)
<input type="checkbox"/>	Attribute Value 7:	0	level7	Utility: (0 ; 0 ; 0)
<input type="checkbox"/>	Attribute Value 8:	0	level8	Utility: (0 ; 0 ; 0)

View

1.0
0.5
0.0

0 0 0 0 0 0 0 0

OK Save Cancel

Figure 4.25. Discrete values utilities

When clicking the “View” button the values entered can be observed in the graph

below (Figure 4.26). If dissatisfied with these values, the user can change the utility ranges and attribute values and click the **View** button again to view the changes. Observe that the discrete attribute values must be increasing as well as their respective average utilities, which are automatically displayed.

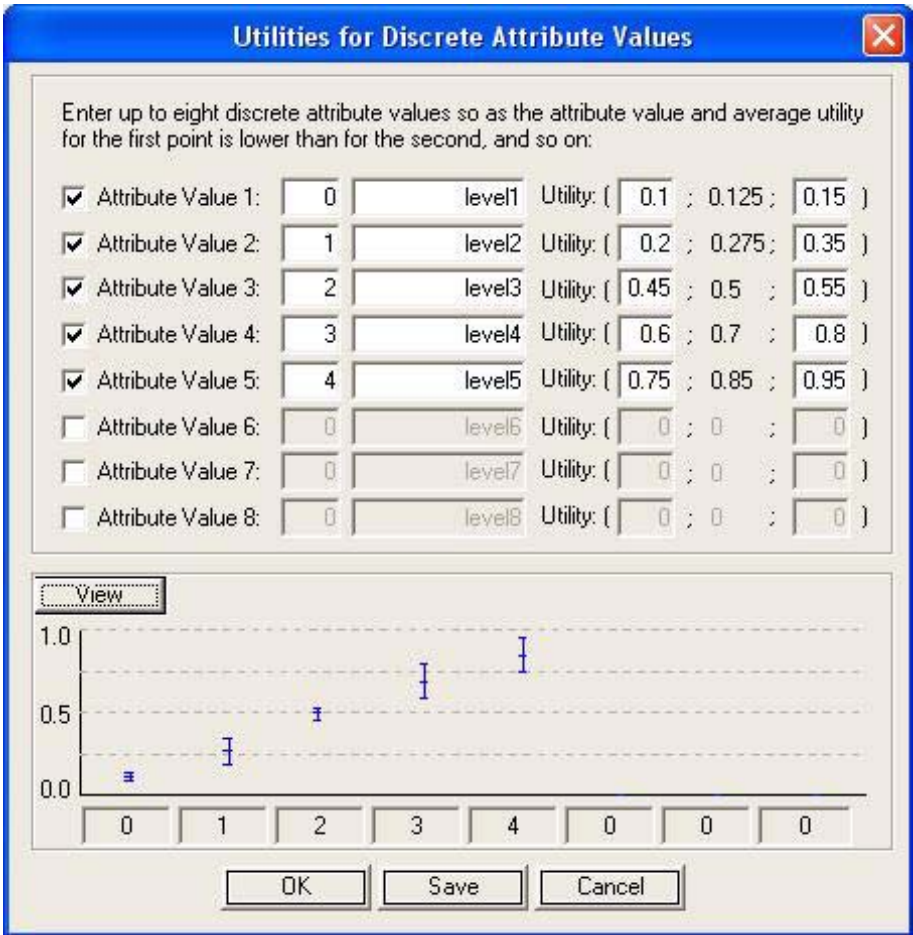


Figure 4.26. Viewing discrete values and imprecise utilities

The discrete values, labels and associated imprecise utilities can be saved by just clicking the **Save** button (the saved utilities can be viewed in Figure 4.27). This means that they will be used by the system in the evaluation process.

Note that alternative consequences may become out of range due to the modification of component utilities. The system will report this situation to the DM when trying to display the alternative ranking or performing sensitivity analysis.

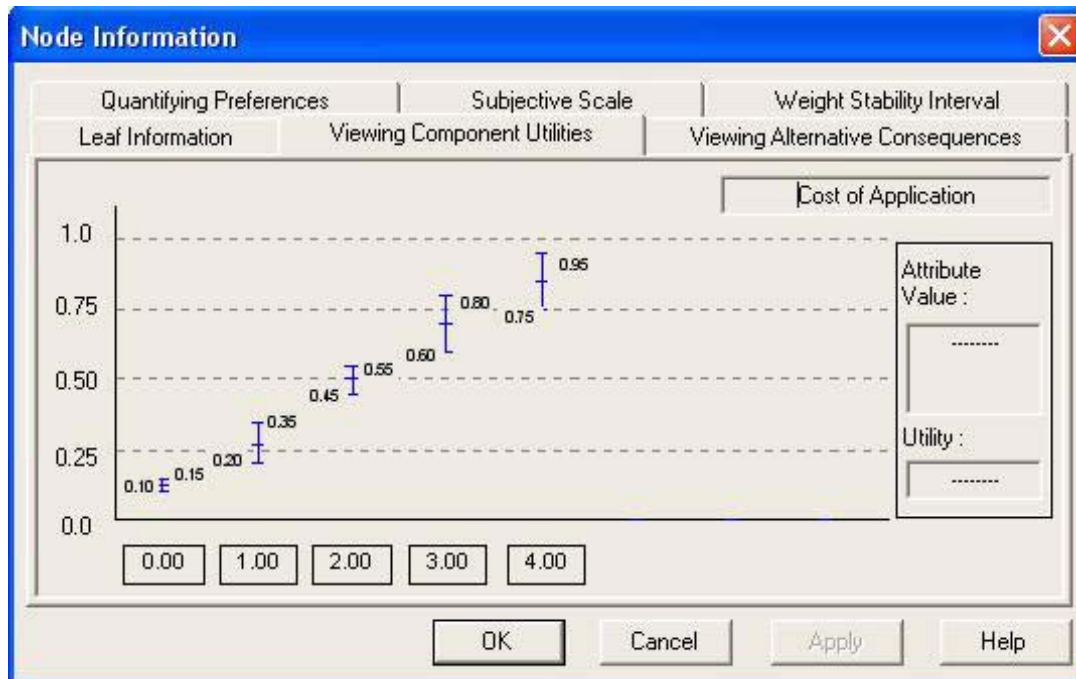


Figure 4.27. Saved imprecise utilities for discrete attribute values

4.1.4 Direct Assignment using Subjective Scales

The DM can decide to use subjective values for one or more leaves of the tree instead of a utility function or imprecise utilities for discrete attribute values and alternative consequences. He/she indicated which attributes have a subjective scale using the **Using Subjective Scale** check button in the respective **Node Information** window when building the objectives hierarchy.

The ranges of subjective values are entered manually through scrollbars depending on the active radio button. When the number of loaded alternatives is greater than ten, the **Next>>** button is activated and can be used to enter the remaining values, as shown in Figure 4.28. Remember that subjective values can only be modified in this window.

If the DM changes his/her mind, clicking this check button again will deactivate the subjective scale for this attribute, and component utilities and new alternative consequences must be entered.

Figure 4.28. Subjective Values

4.2 Weight Elicitation

To get the additive value function, we also need the positive weights w_i , used to add up the separate contributions of all the attributes to get the total utility. Attribute weights reflect the relative importance of the change in the attribute from the worst attribute level to the best attribute level, and can be assessed by multiplying the local weights in the path from that overall objective until each attribute. That local weights represent the relative importance of intermediate nodes in the objective hierarchy.

As said before, the starting point is equally local weighted objectives. If the DM disagrees with the local objective weights, they can be modified by using one of the two weight elicitation methods provided by the GMAA System: *weight elicitation based on trade-offs* and *direct assignment*. Note that imprecision concerning the DM's responses is allowed in both methods by means of ranges of responses to the probability question that the DM is asked. A normalization process is automatically performed from the DM's responses, leading to an average normalized weight and a normalized weight interval for each sub-objective under consideration.

Weight elicitation begins with the attributes and then continues in ascending order through the hierarchy. The first method, perhaps more suitable for the low-level objectives in the hierarchy because it involves a more specific area of knowledge, is based on trade-offs among the respective attributes of the lowest-level objectives stemming from the same objective, Keeney and Raiffa (1976). The DM is asked to give an interval of probabilities such that he/she is indifferent with respect to a gamble and sure consequences.

On the other hand, direct assignment is perhaps more suitable for the possibly more political upper level objectives. The DM has to directly provide a weight interval for each sub-objective under consideration.

4.2.1 Weight Elicitation Based on Trade-offs

To change the weights in the branches of a node, the user must click the button for this node in the tree and select the **Weight Elicitation** folder, and the window shown in Figure 4.29 will be displayed. The weight elicitation based on trade-offs can be used by clicking the **Elicitation** button in the **Weight Elicitation Based on Trade-offs** area.

Node Information

Node Information | Viewing Weights | **Weight Elicitation** | Alternative Classification

Weight Elicitation Based on Tradeoffs

Elicitation { ..., P ; ..., 1-P } equivalent to ...

Direct Assignment

Specify the lower and upper bounds for each one of the sons:

Environmental Impact	Lower Bound: 0	Upper Bound: 0
Social Impact	Lower Bound: 0	Upper Bound: 0
Economic Impact	Lower Bound: 0	Upper Bound: 0
	Lower Bound: 0	Upper Bound: 0
	Lower Bound: 0	Upper Bound: 0

OK Cancel Apply Help

Figure 4.29. Weight Elicitation

This method is mainly used for the lower levels of the tree, and the DM is asked to provide a probability interval $[p_{min}, p_{max}]$ such that he/she is indifferent between a lottery and a sure consequence for each branch. Then, the system will calculate a normalized average weight and a normalized weight interval for each subobjective under consideration by means of a normalization process.

In Figures 4.30 and 4.31, an example can be seen for the objective *Health Impact*.

	P	1-P	
D. Cr. Indiv --->	(0.76000 , 2.47000)	~ 1.61500	
Collective D --->	(20.30000 , 72.30000)	~ 72.30000	

P min. : 0 P max. : 0

<<Back Cancel Next >>

Figure 4.30. Example of weight elicitation based on tradeoffs

	P	1-P	
D. Cr. Indiv --->	(0.76000 , 2.47000)	~ 2.47000	
Collective D --->	(20.30000 , 72.30000)	~ 72.30000	

P min. : 0 P max. : 0

<<Back Cancel Finish

Figure 4.31. Example of weight elicitation based on tradeoffs

In the first probability question the DM has to provide a probability interval in such a way that he/she is indifferent between a lottery, in which we have the most

preferred values for attributes *Dose to Critical Individual* and *Collective Dose* with the demanded probability and the least preferred values for the same attributes with the complementary probability, and sure amounts for the attributes under consideration. Specifically, the mid-value of the attribute range in *Dose to Critical Individual* and the least preferred attribute value for *Collective Dose*. In the second probability question the lottery is the same, but now we have as sure amounts the mid-value of the attribute range in *Collective Dose* and the least preferred attribute value for *Dose to Critical Individual*.

The new normalized average values can be seen in Figure 4.32. These weights have been obtained after entering the values $[0.22, 0.32]$ and $[0.54, 0.67]$ in the **Weights Elicitation Based on Trade-offs** window (Figures 4.30 and 4.31), respectively. The end-points of the respective normalized weight intervals can be watched by using the **Lower bound** and **Upper bound** radio buttons.

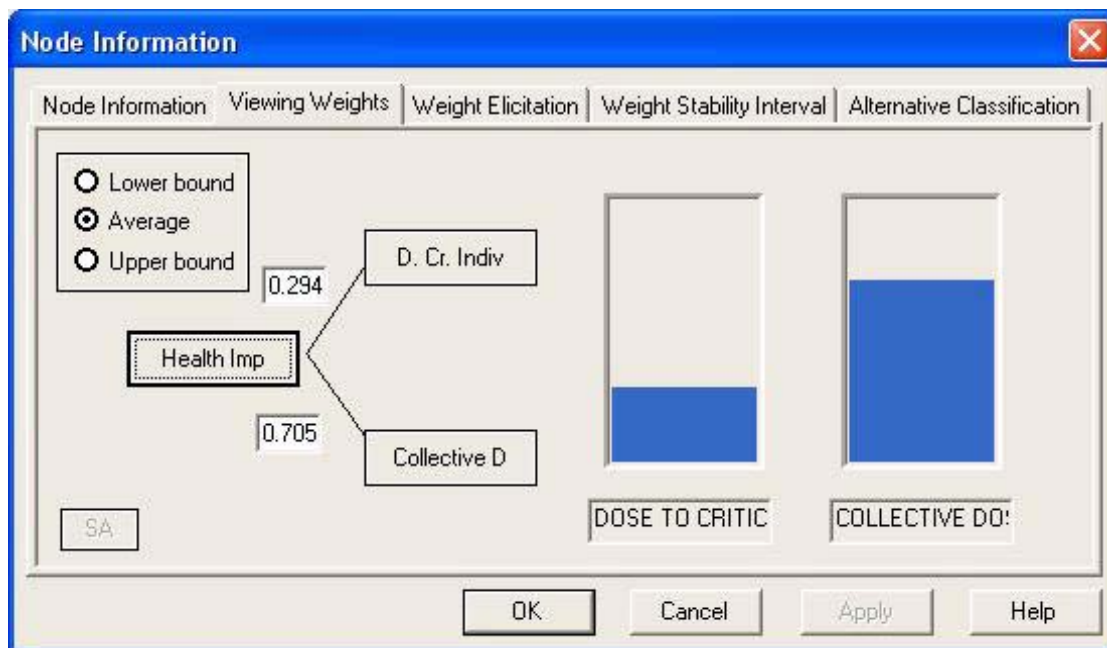


Figure 4.32. Viewing the new normalized weight values for the example

As mentioned above, the weight assessment begins with the attributes and then continues in ascending order through the hierarchy. In an intermediate level, the lotteries will include the best and least preferred values for all the attributes stemming from the sub-objectives under consideration. Moreover, the assessment of average normalized weights and normalized weight intervals will depend on the ones corresponding with

the sub-objectives stemming from them. As a consequence, we should not be surprised if, when reeliciting weights in a lower level, the objectives in the upper ones, in which the method based on trade-offs was used, change as well, see Keeney and Raiffa (1976).

4.2.2 Direct Assignment

This procedure is perhaps more suitable for upper level objectives that could be more political. To use the option we need to focus on the **Direct Assignment** area in the window shown in Figure 4.29. In this example window, the node clicked was *Overall Objective* which has three sons: *Environmental Impact*, *Social Impact* and *Economic Impact*. The DM must directly enter an interval (within which he/she considers the weight to fall) for each son, see Figure 4.33. When the DM clicks the **OK** or **Apply** button, the system will calculate a normalized average weight and a normalized weight interval for each subobjective under consideration by means of a normalization process.

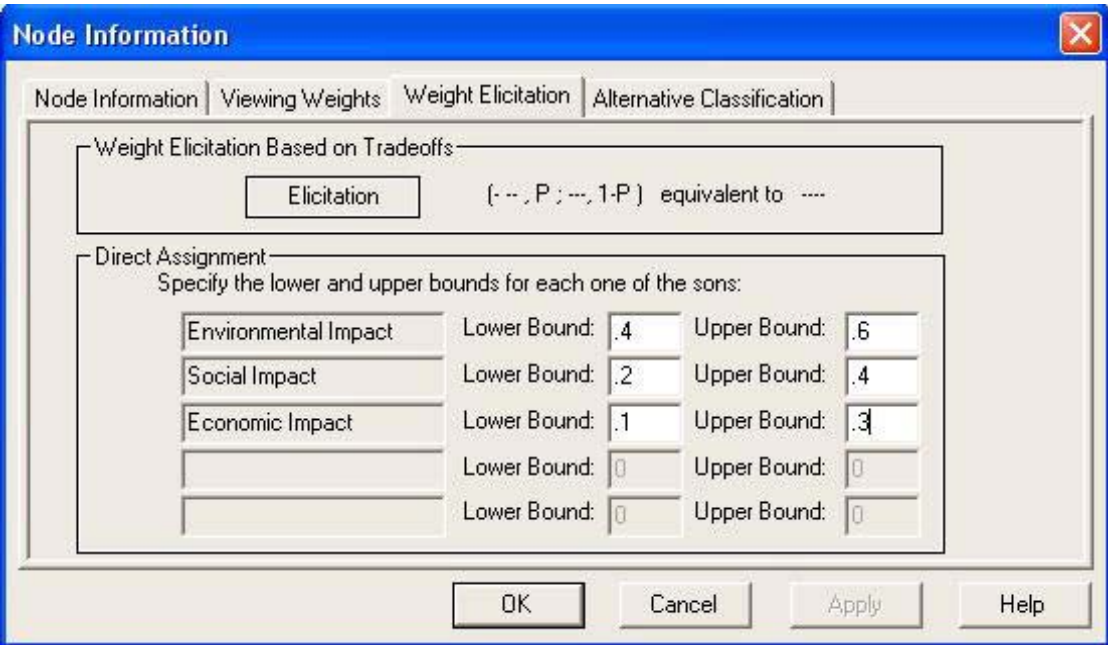


Figure 4.33. Direct Assignment

The average normalized weight that appear in the following window (Figure 4.34) match the weight intervals provided by the DM in Figure 4.33.

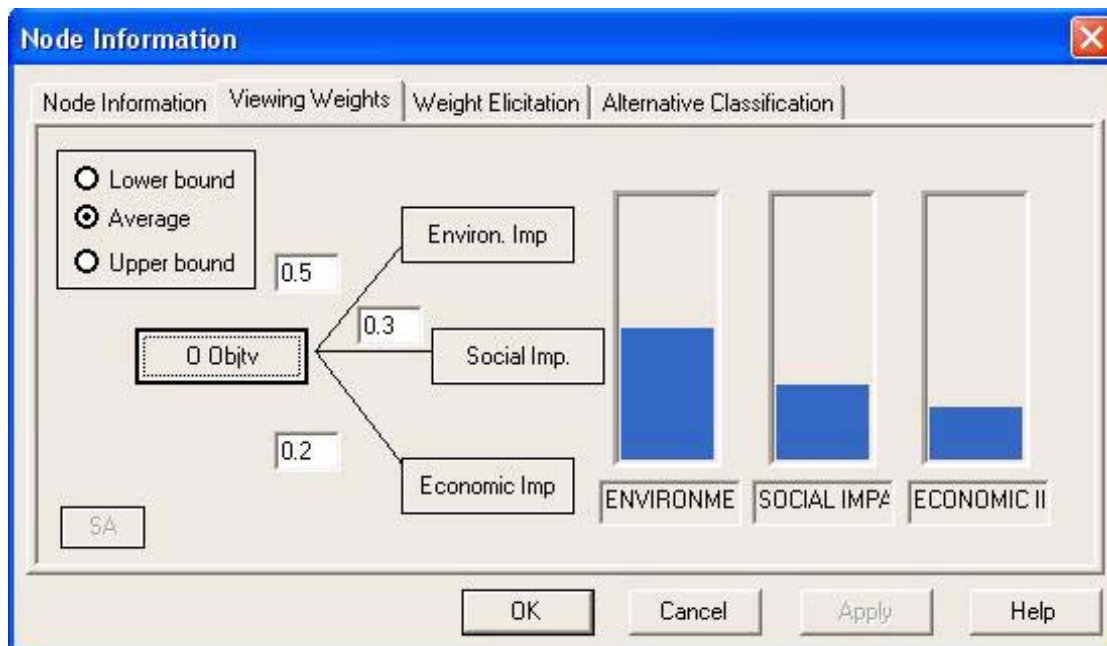


Figure 4.34. Viewing the average normalized weights

Once the relative importance of the objective and attributes has been rated along the branches of the hierarchy, the attribute weight can be assessed by multiplying the respective average weights and normalized weight interval bounds of the objectives in the path from the root (global objective) to each leaf (attribute), see Figure 4.35.

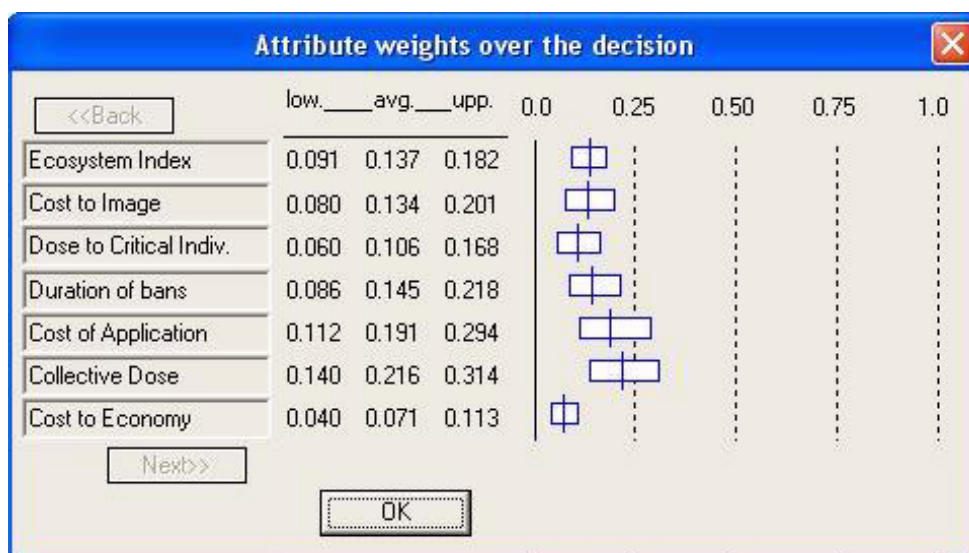


Figure 4.35. Attribute weights

5 EVALUATION OF ALTERNATIVES

Once the DM's preferences have been quantified the different alternatives under consideration can be evaluated by means of an additive multiattribute utility function, whose appearance is

$$\mathbf{u}(S_q) = \sum_{j=1}^n w_j u_j(x_j^q) \quad (1)$$

where w_j is the attribute weight over the decision for the j -th attribute, obtained by multiplying the respective weights of the objectives in the path from the root (global objective) to the j -th leaf (attribute), x_j^q is the consequence for alternative S_q in the j -th attribute and $u_j(x_j^q)$ is the utility associated to the above consequence. For the reasons described in (Raiffa, 1982) and (Stewart, 1996), we consider (1) to be a valid approach.

The additive model is used to assess on the one hand average overall utilities, on which the ranking of alternatives is based and, on the other hand, minimum and maximum overall utilities, which give further insight into the robustness of such ranking. Average overall utilities are obtained by taking into account the mid-points of the uniformly distributed consequence intervals in the respective attributes, their respective average component utilities and the average normalized attribute weights over the decision. To assess the minimum overall utilities the system takes the lower end-points of the imprecise attribute weights over the decision, the lower end-point of the consequence intervals if the respective component utility function is increasing, or the upper end-point if it is decreasing, and the lower utilities in the imprecise utilities corresponding to the above consequences.

The set of alternatives are evaluated and ranked automatically and can be displayed directly either from the **View Alt. Classification** option in the **View** menu or the respective button in the toolbar. The system provides a graphical representation with bars, including overall utilities and ranking, see Figure 5.1. The yellow vertical lines represent the average utilities (used to make the ranking), while the rectangles are bounded by the minimum and maximum utilities.

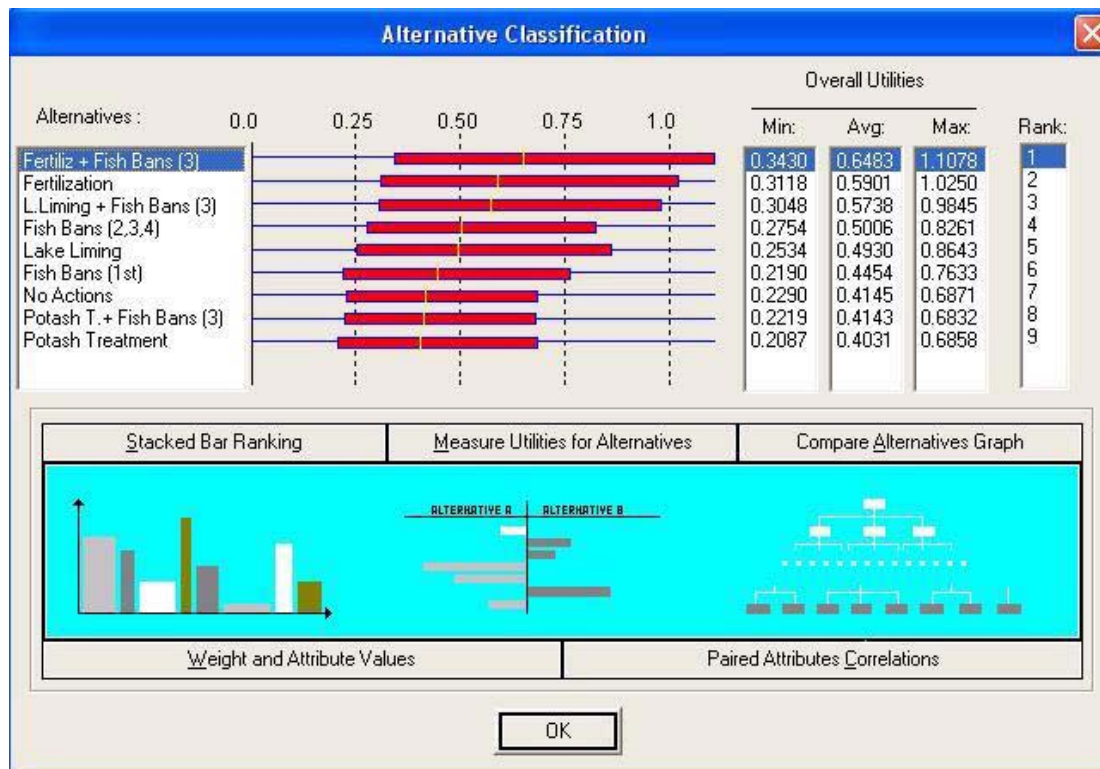


Figure 5.1. The ranked alternatives with their utilities

In some cases, the information obtained through the alternatives evaluation is not meaningful enough so as to definitively recommend an alternative, i.e., we get overlapped imprecise overall utilities, see Figure 5.1. In these cases, the assessment of non-dominated and potentially optimal alternatives and the utilization of Monte Carlo simulation techniques, explained in the sections 6.1 and 6.3, respectively, play a very important role. Both may provide more meaningful information and a process of iteration can be carried out by tightening the respective imprecise alternative consequences, component utilities and weights, reevaluating the alternatives, dominance and potential optimality and performing the Monte Carlo simulation techniques, until a dominant strategy is found.

This ranking and overall utilities can be saved to file by selecting the **Save Results** option in the **WorkSpace** menu. The appearance of this text file is shown in Figure 5.2.

It is also possible to select another objective to rank by. The user just has to select the **Alternative Classification** folder in the respective **Node Information** window,

which is displayed when right-clicking the node in the tree, see Figure 5.3.

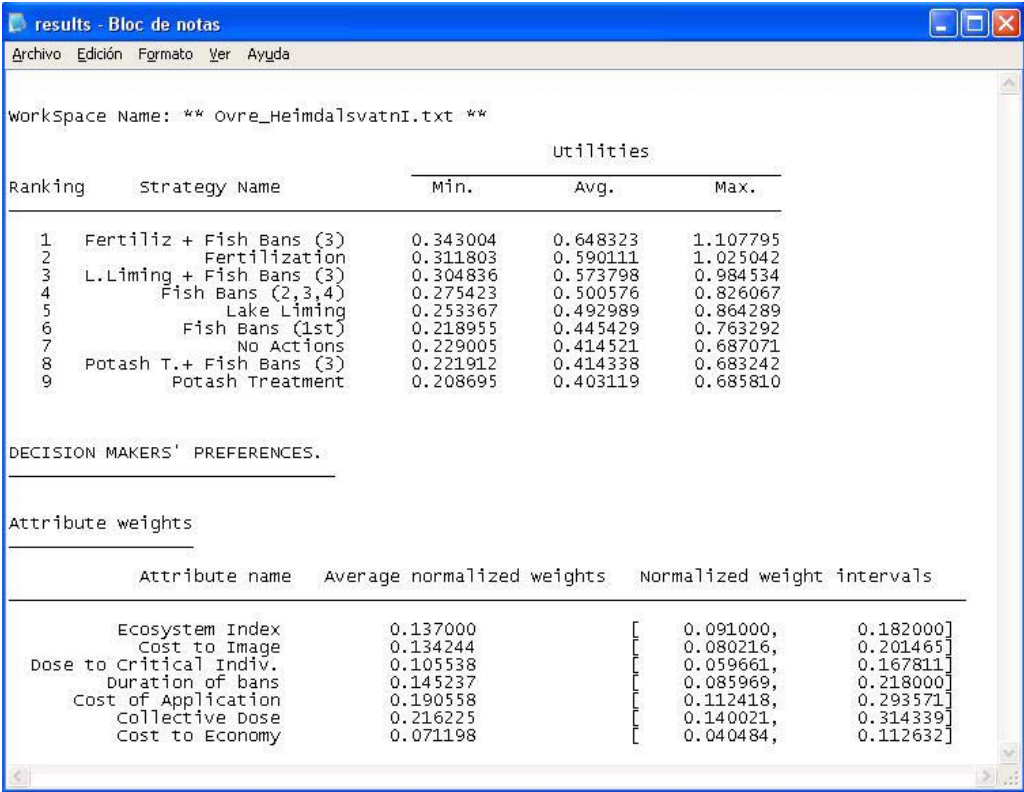


Figure 5.2. Results file appearance

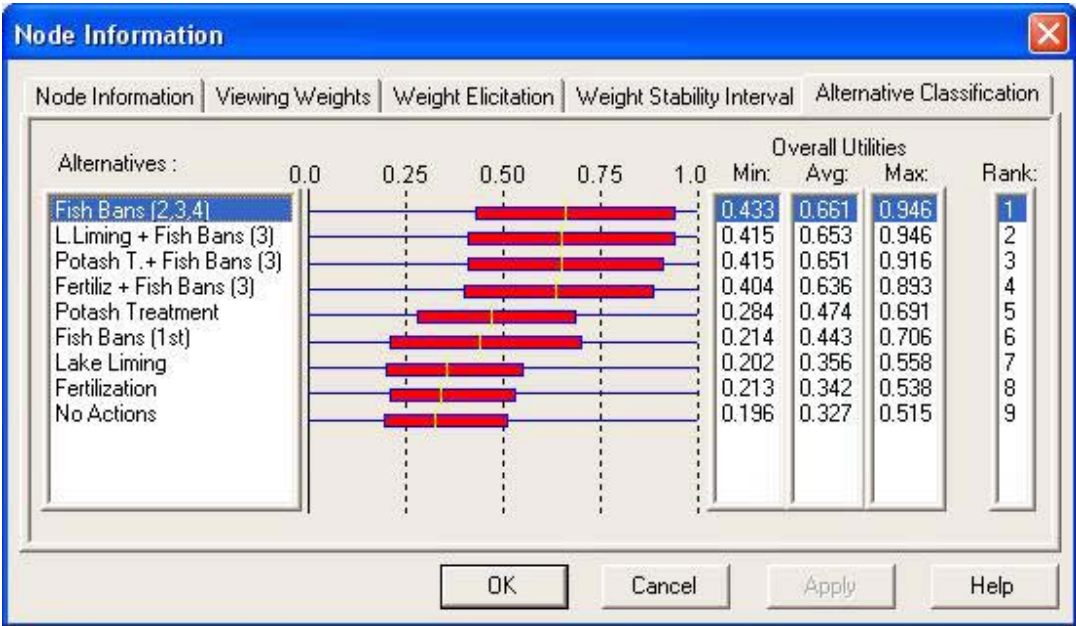


Figure 5.3. Ranking for *Social Impact*

It is also possible to view different displays of ranking results. The following displays are available:

- Stacked Bar Ranking
- Measure Utilities for Alternatives
- Compare Alternatives Graph
- Weight and Attribute Values
- Paired Attributes Correlation

Each of these displays can be selected by clicking on the respective button in the window shown in Figure 5.1.

The *Stacked Bar Ranking* is similar to the alternatives classification, but provides more detail of how the alternative's average utilities for the attributes affect the average utility of the *Overall Objective*, see Figure 5.4.

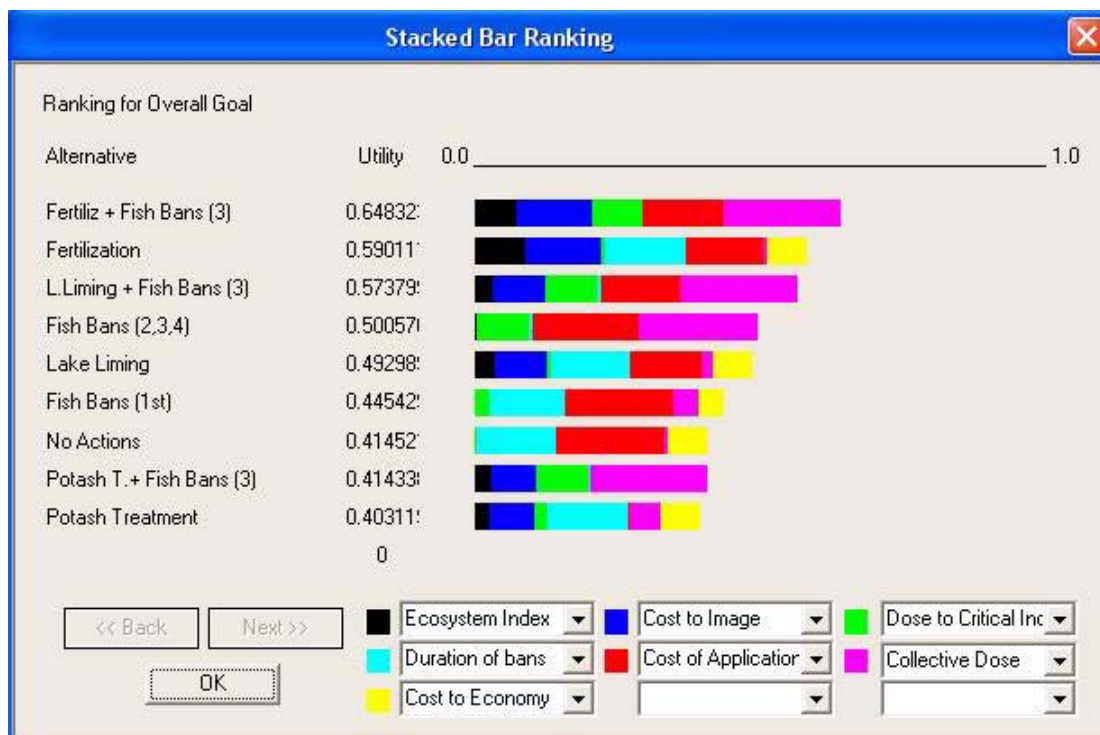


Figure 5.4. Stacked Bar Ranking

In this figure, each alternative has a bar whose length is proportional to its utility in the *Overall Objective*. This bar is made up of other bars that show the influence of the various attributes on the utility result. Not all alternatives will have bars for all attributes, since some alternatives may have a utility of 0.0 (no contribution) for an attribute. If the bar for an attribute is short/long, it means that the alternative performs poorly/well for this attribute.

The **Next>>** and **<<Back** buttons must be used when more than ten alternatives are identified in the workspace. The drop lists contain the attributes corresponding to the different bar colors.

The *Measure Utilities for Alternatives* displays a bar graph showing performance of a single alternative for the attributes, taking into account average consequences and individual utilities. The bar graph is unique in that the width of the bar for an attribute is proportional to its weight, see Figure 5.5.

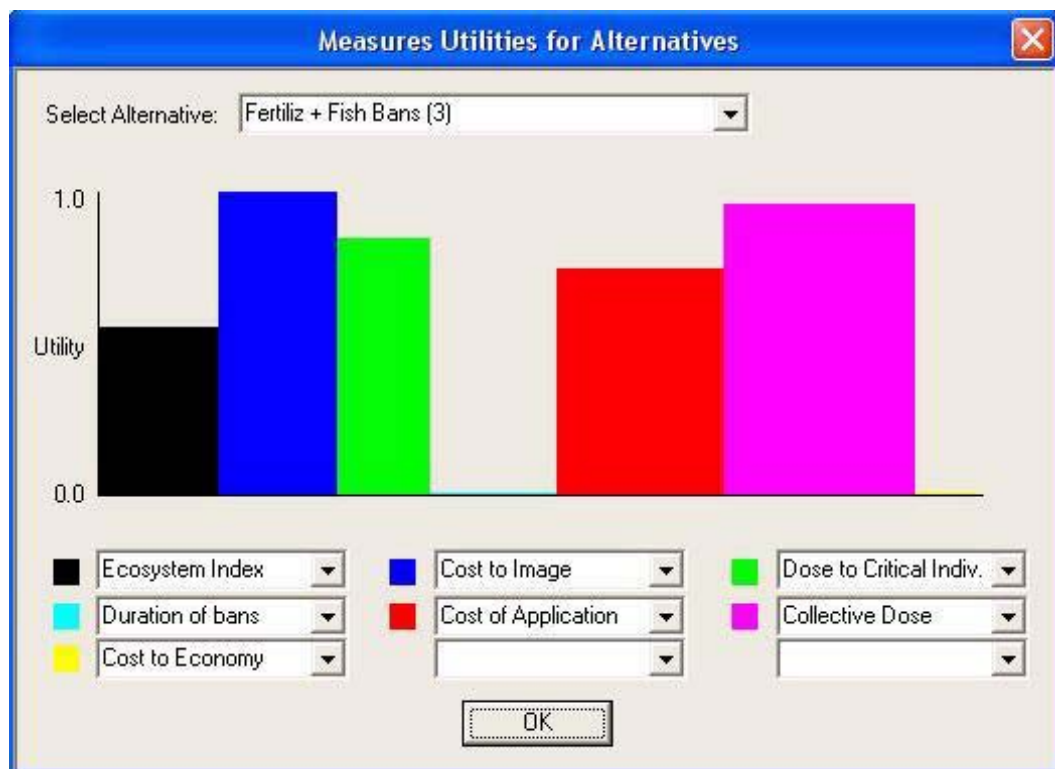


Figure 5.5. Measure Utilities for Alternatives

In this figure, an alternative must be selected from the drop list at the top of the

window. Not all alternatives will have bars for all attributes, since some alternatives may have a utility of 0.0 (no contribution) for an attribute. In the drop lists at the bottom of the window we can see the attributes corresponding to the different bar colors.

Using the *Compare Alternatives Graph*, you can view a detailed comparison of the differences between two alternatives, see Figure 5.6.

First, two alternatives to be compared must be selected from the drop lists at the top of the window. Then, the **Compare Alternatives** button must be clicked.

In the graph, the bars represent measures that favor one alternative over the other, taking into account average utilities. Longer bars indicate more influence on the overall ranking. Bars on the left of the graph favor the alternative with the higher overall ranking. The system draws these bars in blue. Bars on the right of the graph (in red) favor the alternative with the lower overall ranking. The first bar indicates the difference in the overall ranking between the two alternatives.

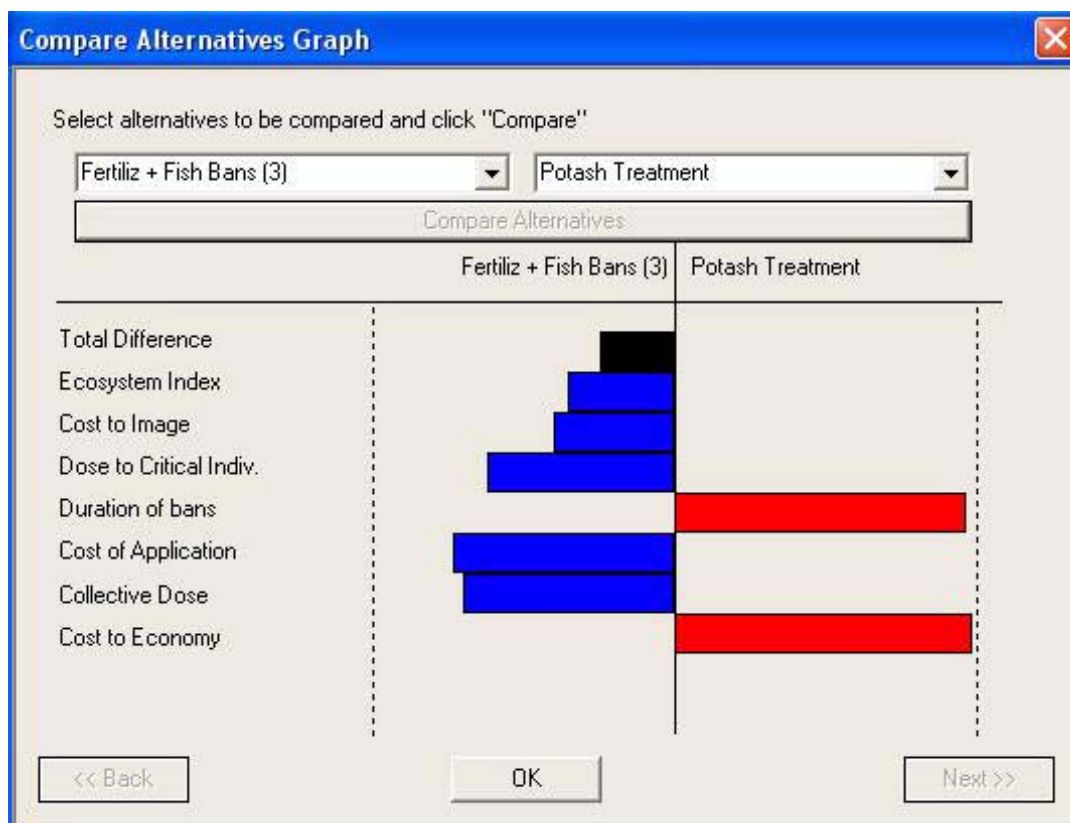


Figure 5.6. Compare Alternatives Graph

By clicking the **Weight and Attribute Values** button (see Figure 5.1), the objectives hierarchy with the assigned average normalized weights for each upper level objective is shown for the alternative selected in the previous window. The display also contains the imprecise consequences of the alternative in question with respect to the attributes (see Figure 5.7 for the alternative *Fertilization + Fish (3 years)*). The system reminds the DM which attributes are using a subjective scale (*Sub. Scale*) or discrete values (*Discrete Value*).

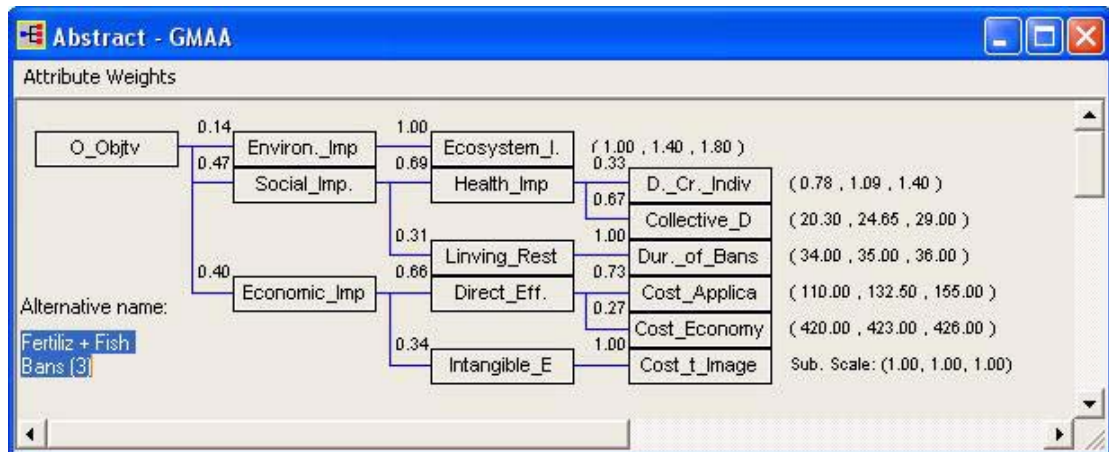


Figure 5.7. The objectives hierarchy including weights and consequences

By clicking the **Attribute Weights** option of the **Attribute Weights** menu in the above window, the normalized interval weights and average normalized weights associated with each one of the attributes in the decision (attribute weights over the decision) are displayed (see Figure 5.8). They are assessed by multiplying the respective average weights and normalized weight interval bounds of the objectives in the path from the root (global objective) to each leaf (attribute).

These weight values are used in the evaluation process described above to output lower, average and upper utilities for each alternative and will also be used later in the Sensitivity Analysis. They are represented both numerically and as a graph. Since the global weights are normalized their sum must obviously be 1.

Next>> and **<<Back** buttons must be used when there are more than ten attributes in the system to view their normalized interval and average normalized weights.

Finally, the *Paired Attributes Correlation* display evaluates/compares alternatives component utilities with respect to pairs of selected attributes. This option is activated

when there are less than 21 alternatives in the Workspace. Therefore, the correlation is not available for 22 or more alternatives. By clicking the **Paired Attribute Correlation** button, the window shown in Figure 5.9 is displayed, where the user must select the two attributes he/she wants to compare.

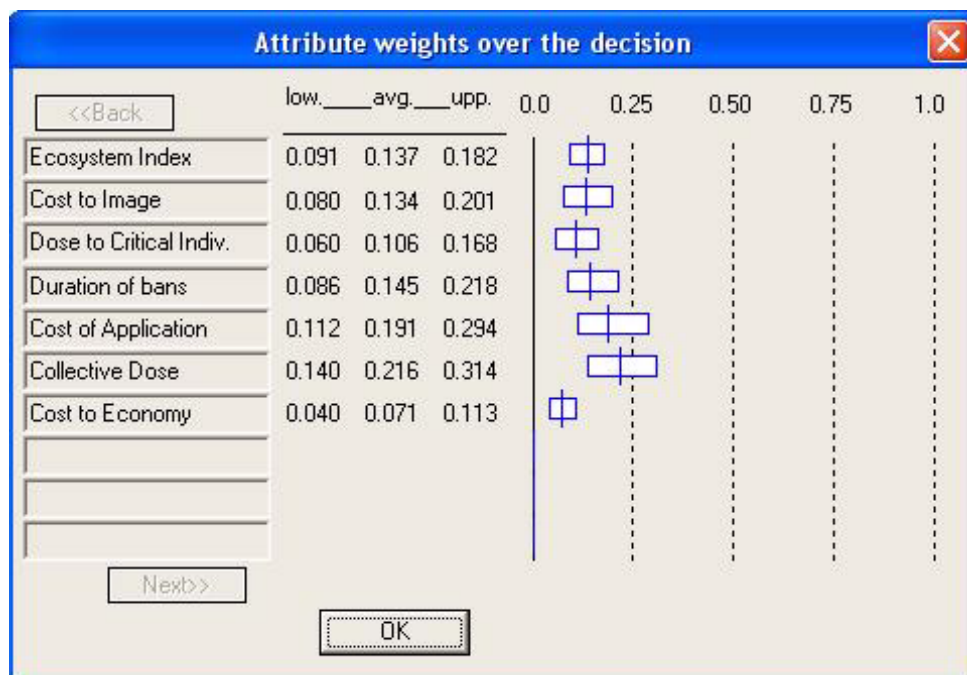


Figure 5.8. Display of the normalized interval and average weights used in the evaluation



Figure 5.9. Selecting attributes for graphical correlation

After selection, a new window is displayed (Figure 5.10), providing a graphical representation of the component utilities resulting for the chosen attributes for the different alternatives under analysis. The colored squares represent each alternative analyzed; the x - and y -axes represent the component utilities for the chosen attributes. Using the radio buttons, the DM can choose to make the correlation using the minimum, average or maximum utilities.

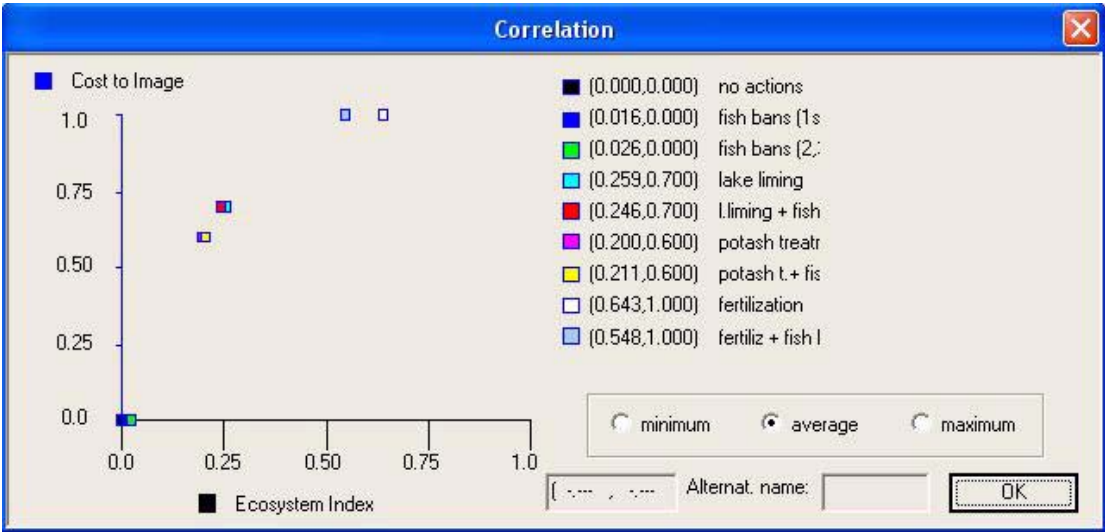


Figure 5.10. Graphical representation of component utilities

6 SENSITIVITY ANALYSIS

The usual way of performing SA involves changing the parameters and observing their impact on the ranking of alternatives, see, e.g., Kirkwood (1997). Hence, if the DM modifies an average normalized weight, normalized weight interval bound, component utility or alternative consequence, the system takes charge of how these changes are propagated through the objectives hierarchy and automatically recalculates the overall utilities for each alternative and the resulting ranking.

The current alternatives classification is shown in Figure 6.1.

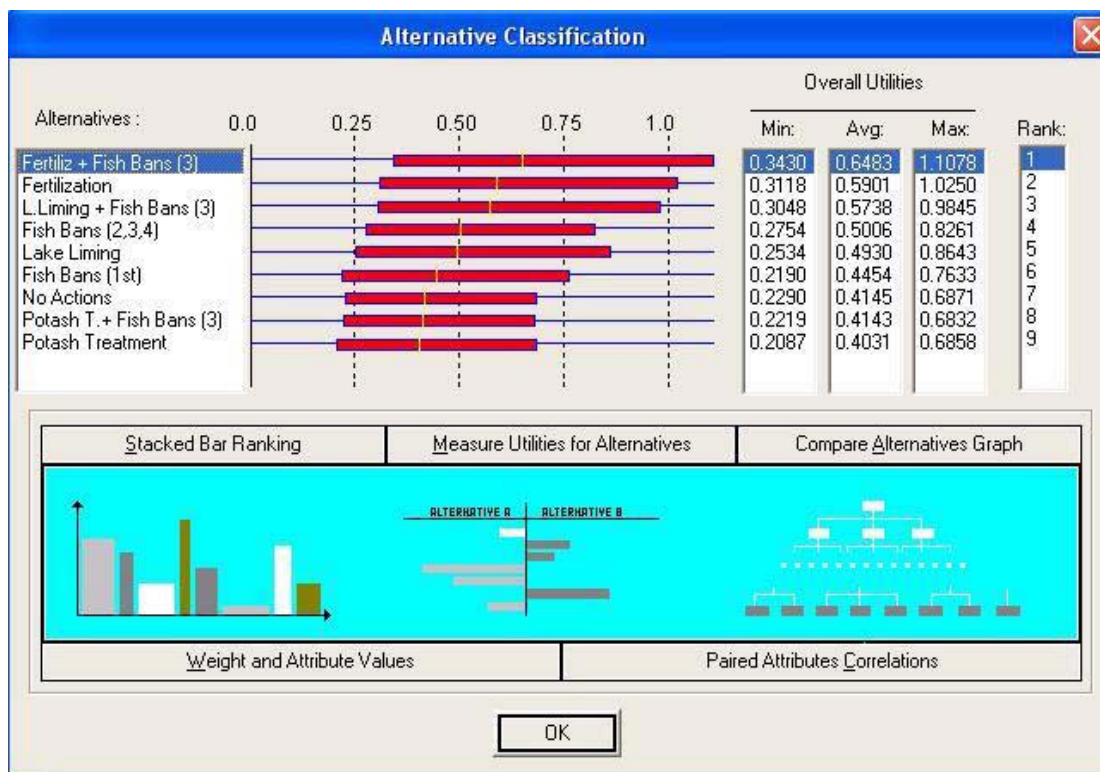


Figure 6.1. The current alternative classification

The procedure for changes to weights is very easy, and is explained by way of an example. Figure 6.2 displays the current weights for objectives stemming from the *Overall Objective*.

The **SA** button is deactivated. However, the user can change any of the average normalized weights or normalized weight interval bounds that appear in this window directly by selecting the respective radio button and writing in the respective box. The

SA button will then be activated and by clicking on this button, the weights will be recalculated as shown in Figures 6.3 and 6.4, where the average normalized weight assigned to the *Environmental Impact* objective has been changed, and the other weights have been automatically readjusted.

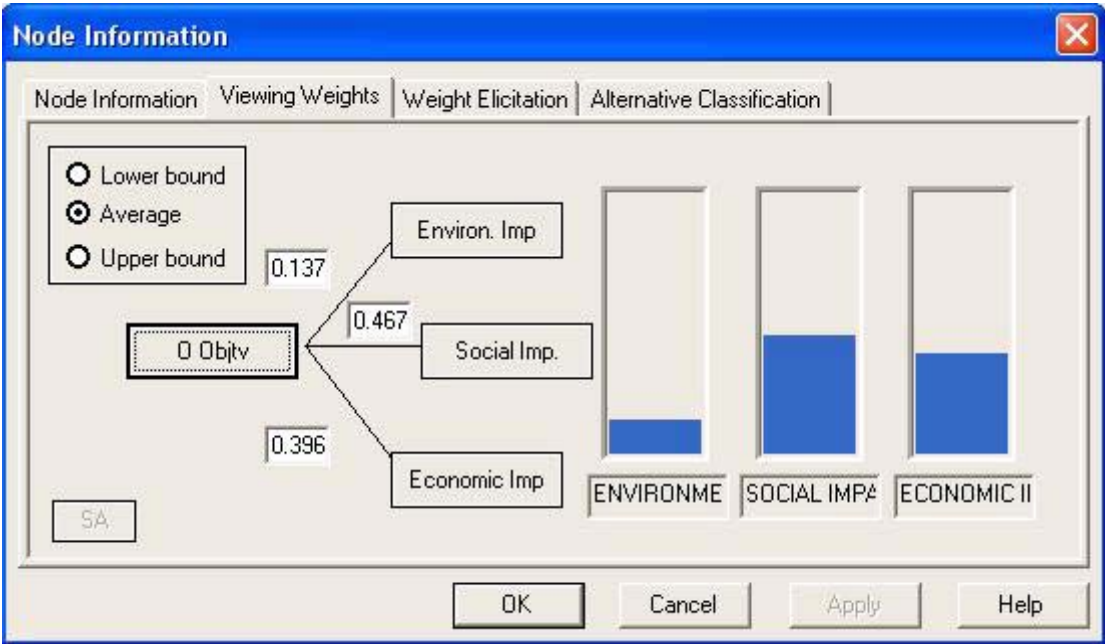


Figure 6.2. Change the weight 0.136

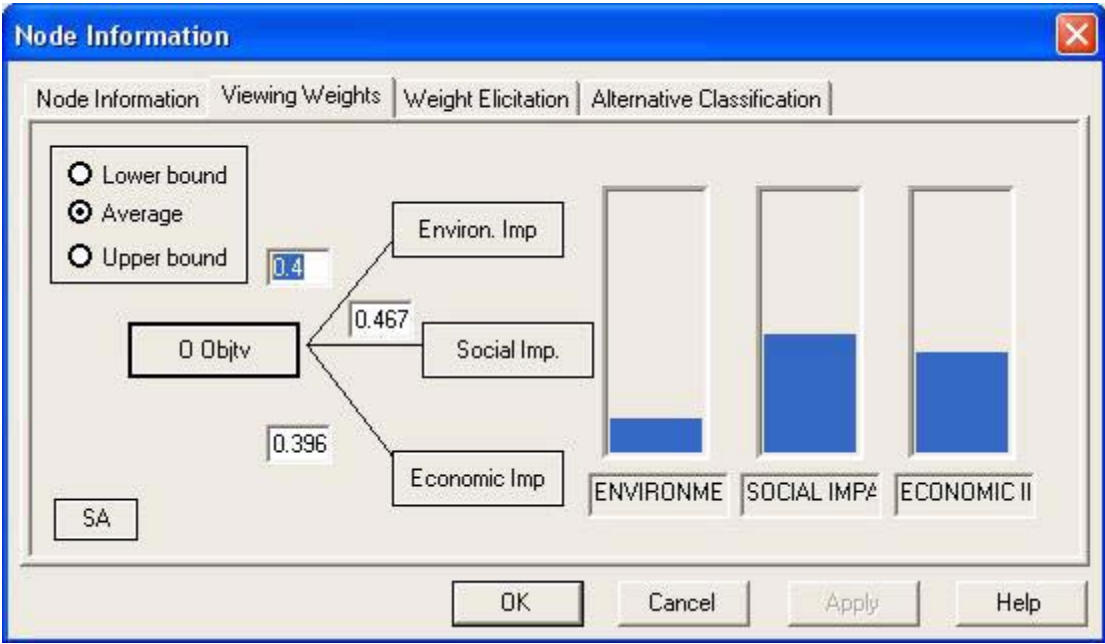


Figure 6.3. Change in weights only takes effect after clicking the “SA” button

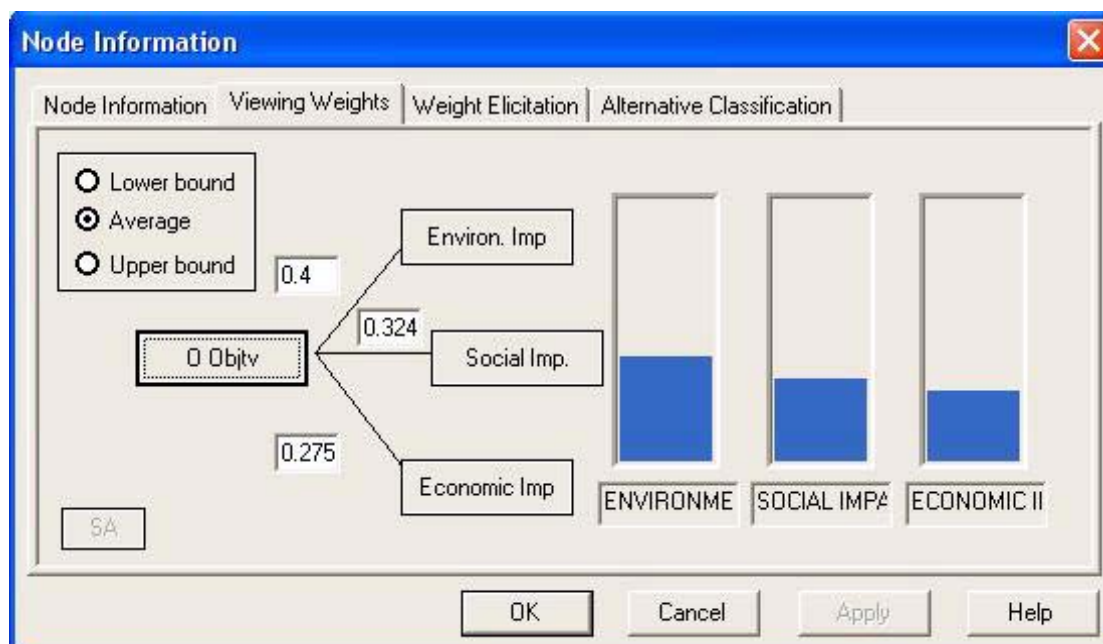


Figure 6.4. New weights

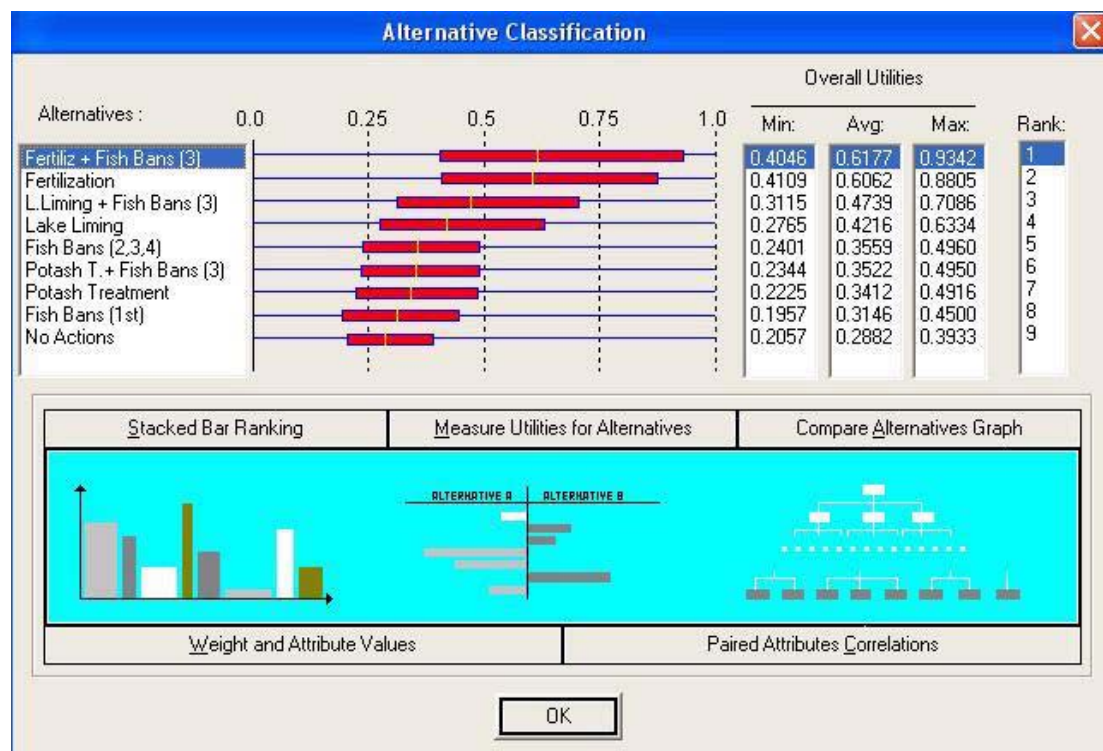


Figure 6.5. The new alternative classification

The system takes charge of how these changes are propagated through the objectives hierarchy and automatically recalculates the overall utilities for each alternative and the resulting ranking. The recalculated results are showed in Figure 6.5.

6.1 Dominance and Potential Optimality

The above SA is a useful but not very systematic to aid for the DM. Therefore, we intend to take advantage of the useful imprecise information collected during the assignment of the component utilities and weights and the entered alternative consequences under uncertainty. Essentially, some more constraints on weights, utilities and alternative consequences can be determined by rough calculations. Then, these can be used in computations described in Ríos Insua (1990) and Ríos Insua and French (1991) to reject definitely bad strategies, mainly by discarding dominated and/or non-potentially optimal alternatives. We will focus on the potentially optimal alternatives, among the non-dominated, i.e., alternatives that are not dominated by any other one and best ranked for at least one combination of imprecise alternative consequences, component utilities and weights.

For this purpose, let us rewrite

$$u(S_l) = \sum_{i=1}^n w_i u_i(x_i^l)$$

where x_i^l is the consequence for alternative S_l in the i^{th} attribute belonging to its consequences interval, u_i is a component utility in the i^{th} attribute belonging to this imprecise component utility attribute and w_i is an attribute weight in the i^{th} attribute belonging to its normalized weight interval.

We will check whether alternative S_l dominates S_q , by solving the optimization problem

$$\begin{aligned} \min \quad & f_{ql} = u(S_l) - u(S_q) \\ \text{s.t.} \quad & w \in W, u \in U \text{ and } x^l, x^q \in X \end{aligned}$$

If the optimal value $f_{ql}^* > 0$, then S_l dominates S_q , and we discard alternative S_q . The main thrust of the above problem is to order the alternatives in a Pareto sense.

We can also determine potentially optimal (p.o.) alternatives, i.e., those S_j that maximize $u(S_j)$ for some $w \in W$, $u \in U$ and $x^j \in X$ as solutions. The optimization problem to be solved to determine if the alternative S_j is p.o. is:

$$\begin{aligned} \min z = & f_i \\ \text{s.t.} \quad & \sum_{i=1}^n w_i u_i(x_i^j) - \sum_{i=1}^n w_i u_i(x_i^l) + f_j \geq 0 \quad \forall l \neq j \\ & w \in K, u \in U \quad \text{and} \quad x^j, x^l \in X \end{aligned}$$

If the optimal value $f_j \leq 0$, then alternative S_j is potentially optimal.

The above optimization problems are not linear but they can be transformed into linear problems and solved using the Simplex Method, see Mateos *et al.* (2003).

To view non-dominated and potentially optimal alternatives, we must select the **Dominance/Potential Optimality** option in the **Sensitivity Analysis** menu or press the respective toolbar button, which will display the window shown in Figure 6.6.

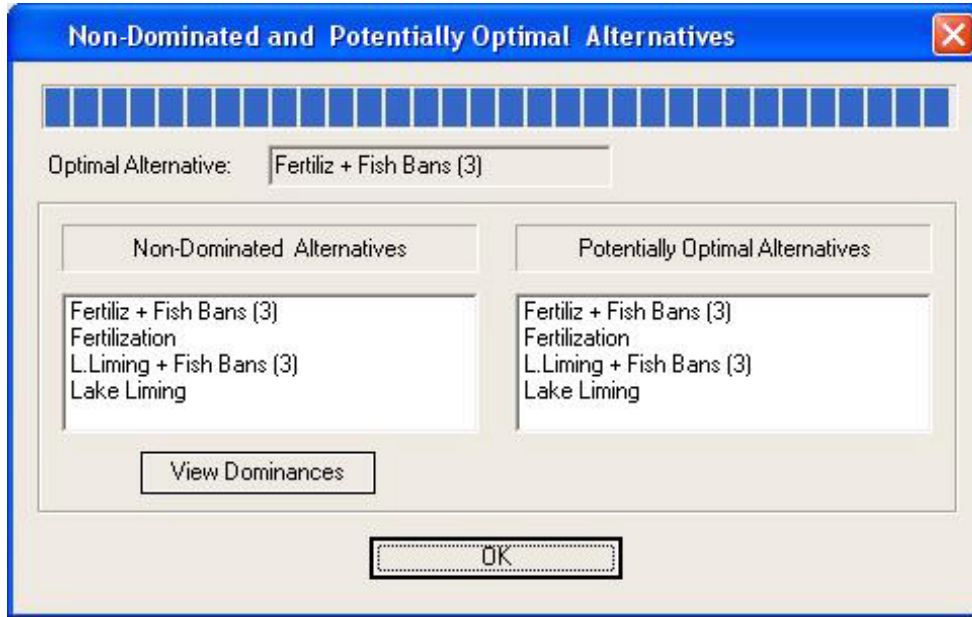


Figure 6.6. Non-Dominated and Potentially Optimal Alternatives

Further details about the dominance between alternatives can be viewed by clicking the **View Dominance** button, see Figure 6.7.

The position of the alternative in the rows and columns depends on the alternatives ranking in the table shown in this figure, and only the necessary optimization problems are solved, i.e., if the result of the dominance problem is greater than zero the

alternative placed in this column is dominated so it can be discarded. In this example, six alternatives out of the nine considered are non-dominated. Moreover, all the non-dominated alternatives are potentially optimal as well.

	Fertilization	L.Liming + Fish Ba...	Lake Liming	Fish Bans (2,3...	Potash T.+ F
- Fertiliz + Fish Bans (3)	-0.3784	-0.1462	-0.1415	0.0402	0.0244
- Fertilization		-0.2234	-0.0783
- L.Liming + Fish Bans (3)			-0.2354
- Lake Liming			
- Fish Bans (2,3,4)					...
- Potash T.+ Fish Bans (3)					
- Potash Treatment					
- Fish Bans (1st)					

Figure 6.7. Dominance details

As cited above, in some cases, the information obtained from the alternatives evaluation, by means of the additive multiattribute utility model, is not meaningful enough so as to definitively recommend an alternative, i.e., we get overlapped imprecise overall utilities, see Figure 5.1. In these cases, the assessment of non-dominated and potentially optimal alternatives plays a very important role. It may provide more meaningful information and a process of iteration can be carried out by tightening the respective imprecise alternative consequences, component utilities and weights, reevaluating the alternatives, dominance and potential optimality, until a dominant strategy is achieved.

6.2 Weight Stability Analysis

Another way of performing SA involves assessing weight stability intervals. The system includes two possibilities for assessing weight stability intervals. The first is to assess the interval in which average normalized weight for a specific objective can vary without affecting the best alternative. The second is to assess the interval in which average

normalized weight for a specific objective can vary without affecting the alternatives ranking. Again, it is easier to understand if we give an example for the second case. Suppose that the current alternatives ranking as shown in Figure 6.1.

If the **Weight Stability Interval** folder is selected for *Collective Dose*, the system ask us which type of stability interval we want to assess and then a new window specifying an interval is opened, as shown in Figure 6.8. The current value of the average normalized weight for *Collective Dose* branch is 0.672. However, this value could be changed within the specified interval $[0.33, 0.679]$ without implying changes in the ranking of alternatives shown in Figure 6.1, in spite of changes to their respective overall utilities.

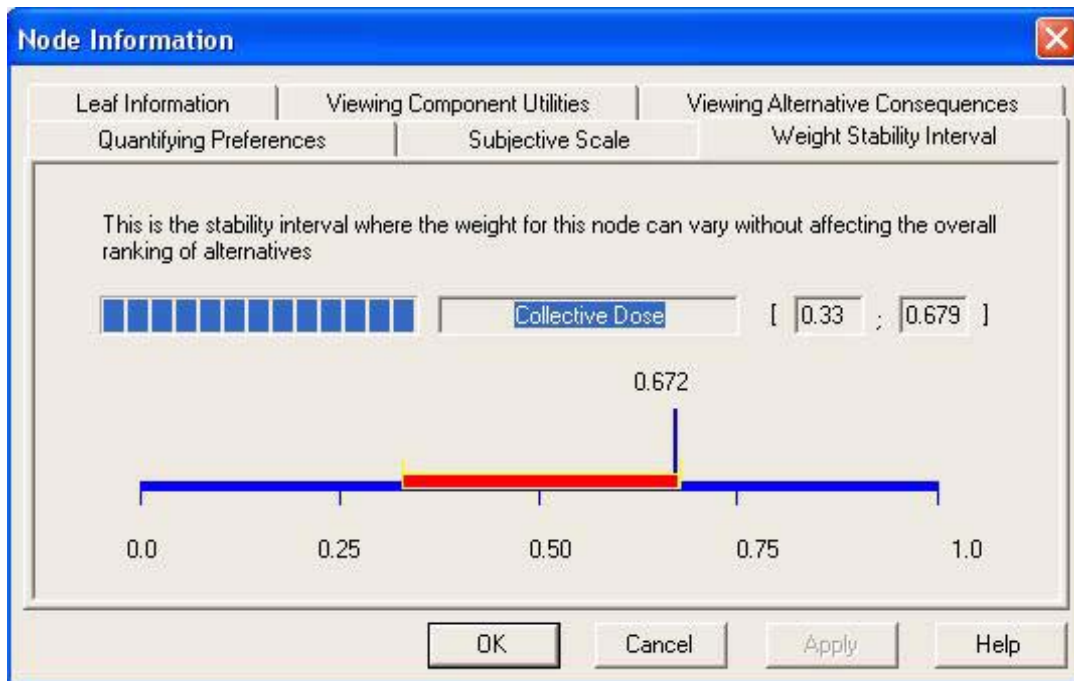


Figure 6.8. Weight Stability interval for *Collective Dose*

If we then enter a new weight value for *Collective Dose* outside the interval, for instance 0.68, (Figure 6.9), a new alternatives ranking is output (shown in Figure 6.10).

Comparing Figure 6.1 and Figure 6.10, we find that the ranking of *No Action* and *Potash Treatment + Fish Ban (3)* have changed, now *Potash Treatment + Fish Ban (3)* is better than *No Action*. Since the new value was outside the interval, the alternatives ranking has changed.

Stability intervals can also be viewed by selecting the **Weight Stability Intervals** option in the **Sensitivity Analysis** menu.

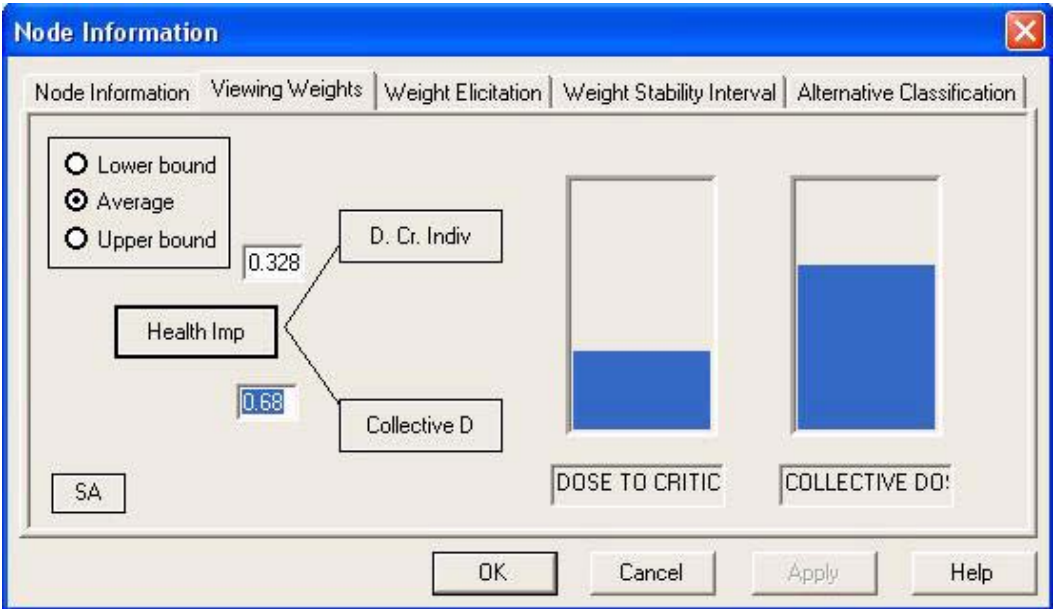


Figure 6.9. Modifying the *Collective Dose* weight

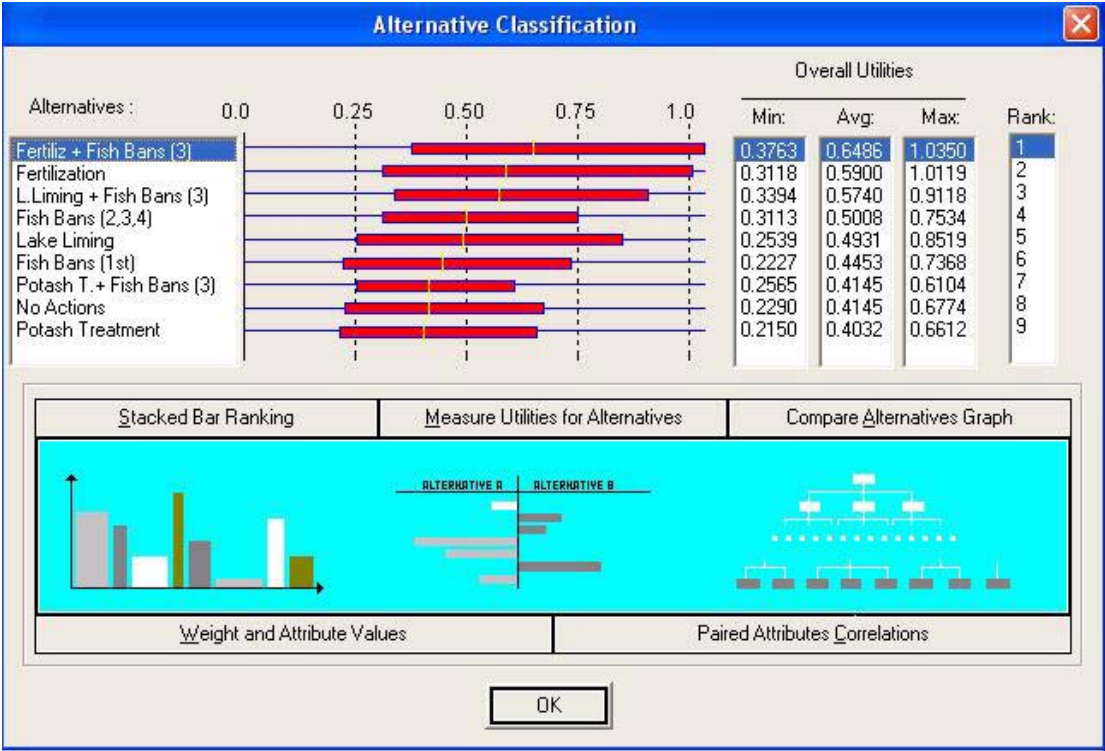


Figure 6.10. The new alternative classification

6.3 Simulation Techniques for SA

This kind of sensitivity analysis uses Monte Carlo simulation and enables simultaneous changes to the weights, generating results that can easily be analyzed statistically to provide insight into the multi-criteria model recommendations, Jiménez *et al.* (2003).

We propose selecting the weights at random using a computer simulation program so that the results of many combinations of weights, including a complete ranking, can be explored efficiently. The system uses a multiplicative linear congruential generator based on Schrage's method, first published in 1979, and later refined in 1983, Bratley *et al.* (1983). It provides a virtually infinite sequence of *statistically independent* random numbers, *uniformly* distributed between 0 and 1.

Three general classes of simulation will be presented: random weights, rank order weights and response distribution weights. If the **Simulation Techniques for SA** option is selected in the **Sensitivity Analysis** menu, the system displays the window shown in Figure 6.11, where one of the classes of simulation must be chosen.



Figure 6.11. Choosing a class of simulation technique

6.3.1 Random Weights

As an extreme case, weights for the measures can be generated completely at random. This approach implies no knowledge whatsoever of the relative importance of the measures. In many multi-criteria settings, the scores of the alternatives significantly limit the subset of potential rankings.

By selecting the **Random Weights** radio button and clicking on the **Next>>** button, the window shown in Figure 6.12 is displayed.

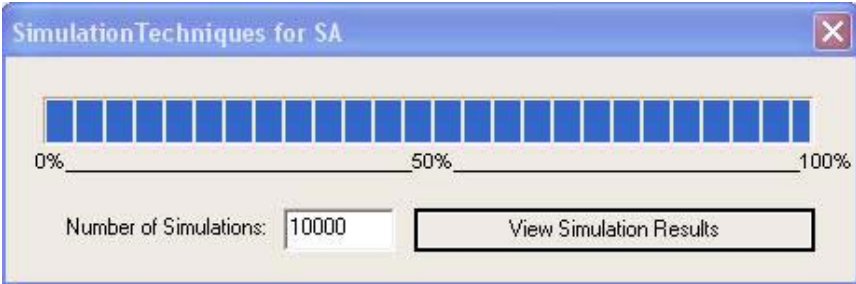


Figure 6.12. Simulation techniques for SA progressing bar

The DM can adjust the number of simulations to be performed, whose default value is 10000. Just by clicking the **View Simulation Results** button, the simulation results window shown in Figure 6.13 is displayed.

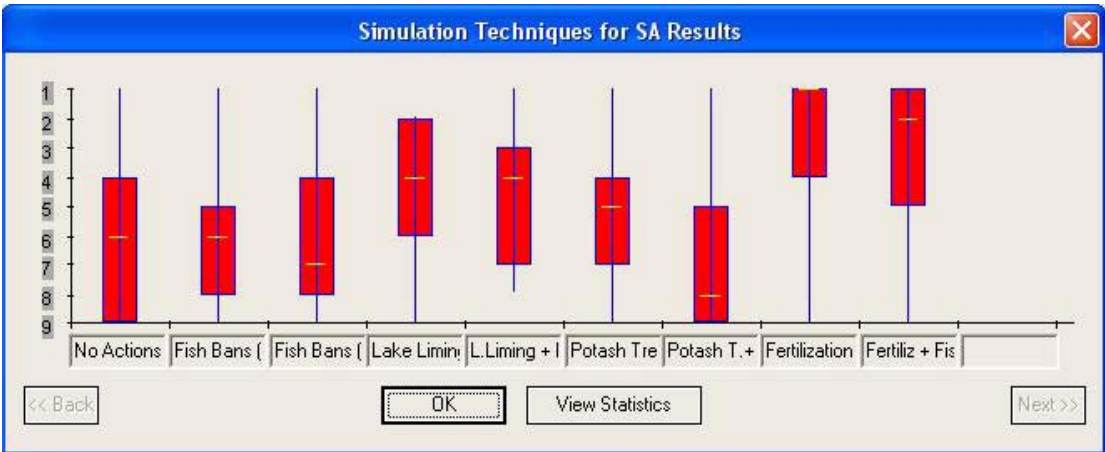


Figure 6.13. Simulation techniques for SA results

In this example window, there are nine alternatives, which are represented along the *x*-axis. The *y*-axis represents positions in the alternatives ranking. A box diagram is assigned to each alternative, whose meaning is shown in Figure 6.14. Statistical values can be viewed by clicking the “View statistics” button, and Figure 6.15 will be displayed.

Taking into account the results obtained from Figures 6.13 and 6.15, the DM may realize that some of the alternatives may be candidates for removal because their best

performances are not good. Figures 6.13 and 6.15 may also prove useful in discussions with the decision makers.

Of course, removing any alternative that has been ranked first in at least one simulation could be dangerous, as there is a combination of weights that led to it being the most preferred.

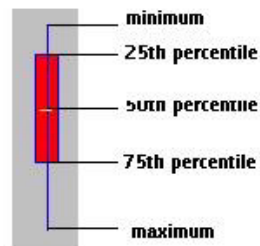


Figure 6.14. Box diagram

	Mode	Min.	25th percentile	50th percentile	75th percentile	Max.	Mean	Std. Deviation
No Actions	9	1	4.000	6.000	9.000	9	6.213	2.457
Fish Bans (1st)	8	1	5.000	6.000	8.000	9	6.038	1.777
Fish Bans (2,3,4)	9	1	4.000	7.000	8.000	9	6.289	2.392
Lake Liming	2	2	2.000	4.000	6.000	9	4.294	2.020
L. Liming + Fish Ba...	7	1	3.000	4.000	7.000	8	4.520	2.068
Potash Treatment	5	1	4.000	5.000	7.000	9	5.487	2.160
Potash T. + Fish B...	9	1	5.000	8.000	9.000	9	6.839	2.154
Fertilization	1	1	1.000	1.000	4.000	9	2.335	1.801
Fertiliz + Fish Bans...	1	1	1.000	2.000	5.000	9	2.986	2.112

Figure 6.15. Statistics

6.3.2 Rank Order Weights

Randomly generating the weights while preserving their criteria rank order places substantial restrictions on the domain of possible weights that are consistent with the DM's judgement of criteria importance. Therefore, the results from the rank order simulation may provide more meaningful results.

The DM can enter the rank order in the window shown in Figure 6.14 by selecting the **Rank Order Weights** radio button in Figure 6.11. The attributes of the objectives hierarchy appear in the list on the left-hand side of the window and the DM can build the complete or partial rank order weights using the **Insert>>>** and **<<<Back** buttons. Logically, the first attribute in the rank order list is the most preferred. The DM can opt to provide a complete rank order but, as this is not always possible, he/she is able to provide a partial rank as well. Once the DM agrees with the rank order, results can be viewed just by clicking the **Next>>** button. Figure 6.12 will be displayed.



Figure 6.16. Rank order weights

6.3.3 Response Distribution Weights

This third type of simulation-based sensitivity analysis recognizes that the weight assessment procedure is subject to variation. For a single DM, this variation may be in the form of response error associated with the weight assessment.

As mentioned in section 3.3, where the weights assignment methods are explained, we assume imprecision allowing the DM to enter intervals, rather than single values in his/her responses. Therefore, normalized weight intervals are defined in the nodes of the objectives hierarchy.

While in the first class of simulation, random weights simulation, attribute weights were randomly assigned values between 0 and 1 (taking into account that the sum of

the whole is the unit), now attribute weights are randomly assigned values taking into account the normalized weight intervals provided by the DM in the weights assignment methods, the attribute weight intervals. The normalized weight intervals used in this simulation can be viewed in the window shown in Figure 5.8. The result of this class of simulation can be viewed by clicking the **Response Distribution Weights** radio button in the window shown in Figure 6.11.

As cited above, in some cases, the information obtained from the alternatives evaluation, by means of the additive multiattribute utility model, is not meaningful enough so as to definitively recommend an alternative. In these cases, the utilization of this third type of simulation-based SA plays a very important role. It may provide more meaningful information and a process of iteration can be carried out by tightening the respective imprecise alternative consequences, component utilities and weights, and performing the simulation, until a dominant strategy is found.

7 HELP, PRINT AND PRINT PREVIEW

The user can get help and information about the program in HTML form by clicking on the **manual.html** icon in the **GMAA** application folder or by clicking the **Start** button, pointing to **Programs**, and then clicking on **Generic Multi-Attribute Analysis**, see Figure 7.1.

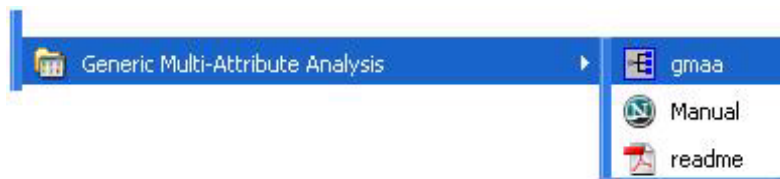


Figure 7.1. Manual index

The information is indexed according to the user's guide units explained above, see Figure 7.2. The DM can open any unit by clicking on the manual, and a new window containing the chosen unit will be displayed, as shown in Figure 7.3 for the *Alternatives Processing* unit. The references (in blue) in the text are links to the *References* Unit. If we click on the reference the system takes us to the respective place in the *References* Unit, see Figure 7.4.

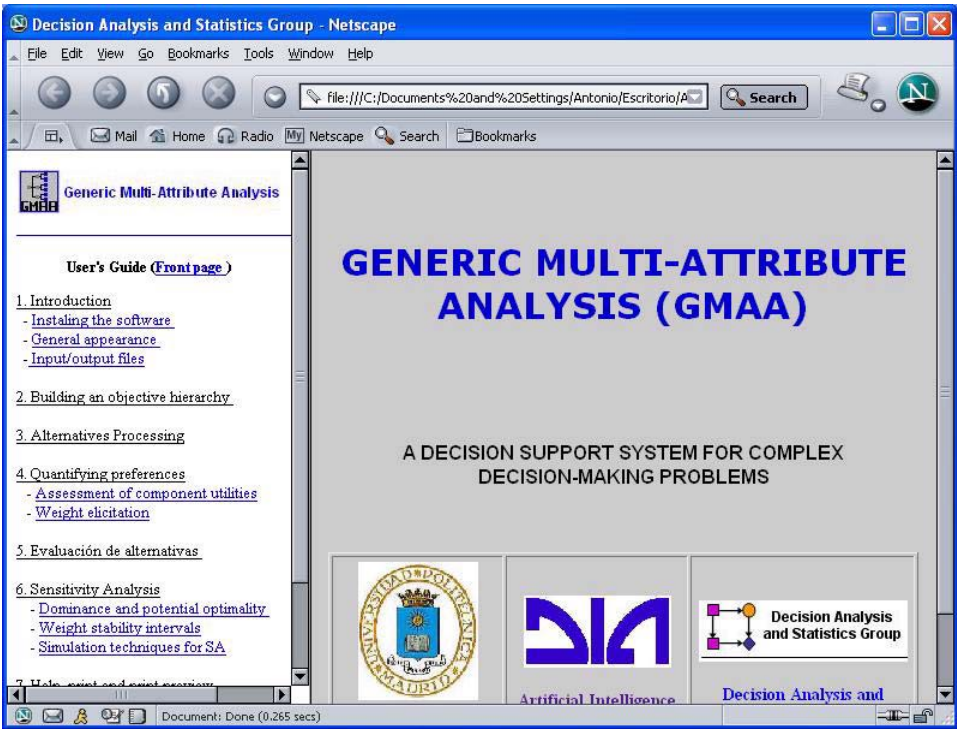


Figure 7.2. User's guide index

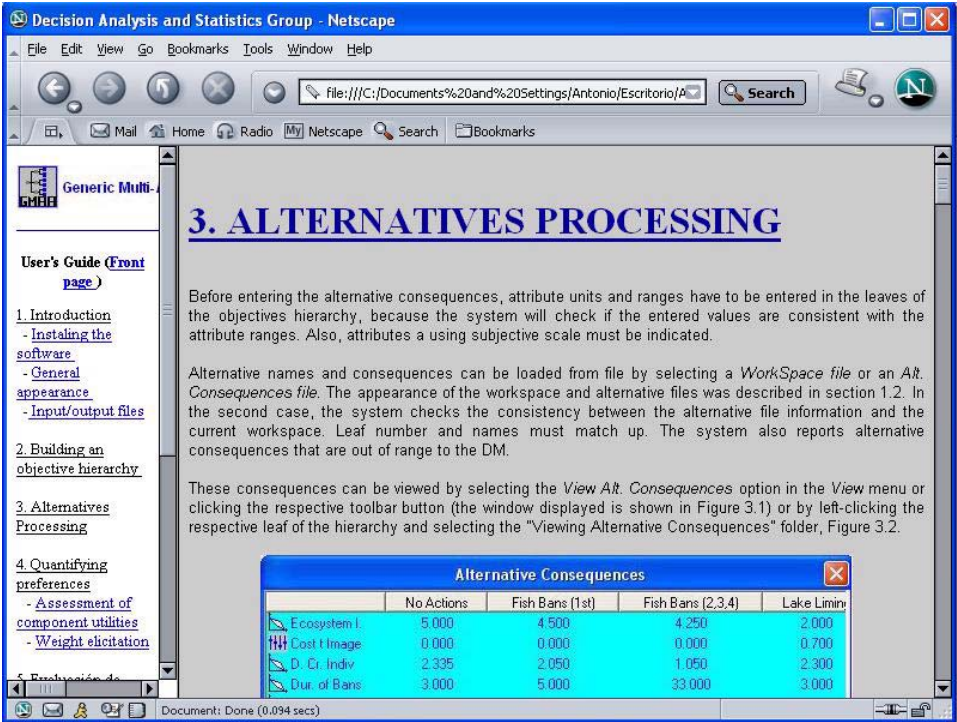


Figure 7.3. Alternatives Processing

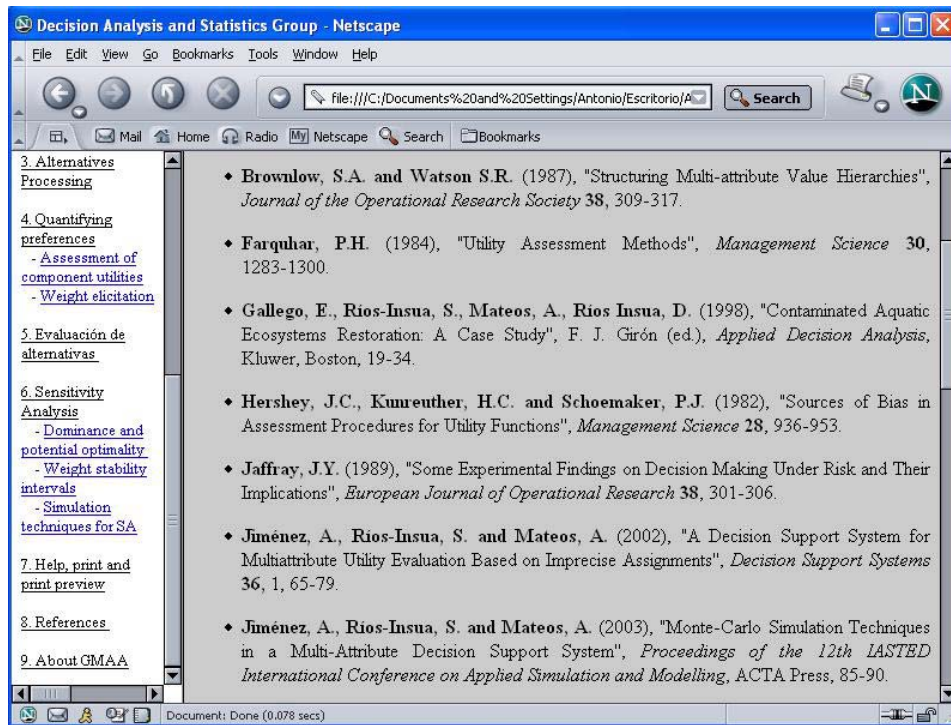


Figure 7.4. (Brownlow and Watson, 1987) reference

The DM can use the **Forward** and **Back** buttons in the Microsoft Internet Explorer or Netscape Navigator toolbar to navigate through the user's manual.

The system offers the possibility of printing the current alternatives ranking and related information from the **Workspace** menu or the respective toolbar button. This document contains the alternatives ranking, information related to the alternative consequences (tables with the consequence intervals for the different attributes) and information related to the attributes (ranges, best and worst values, units and normalized average weights and weight intervals).

The DM can preview the appearance of the document on paper without actually sending it to the printer, see Figure 7.5. This option provides the DM with more information than the **Save Results to File** option in the **File** menu, which only saves the alternatives ranking.

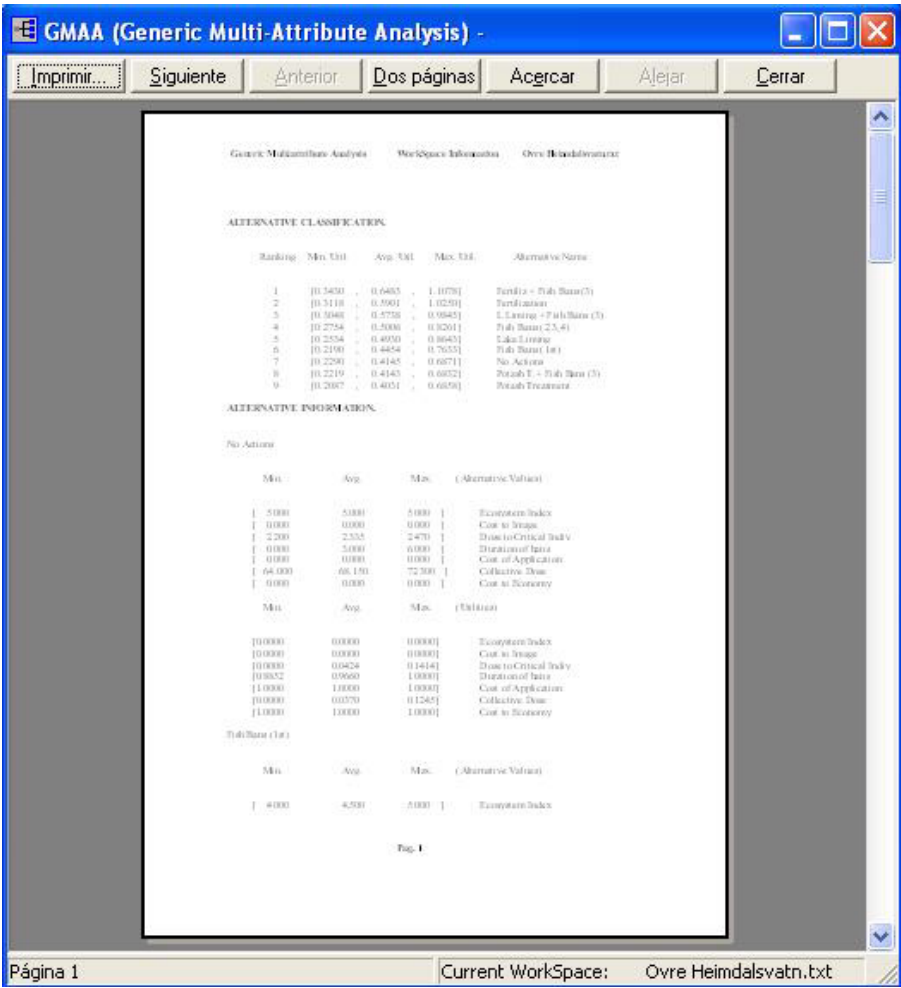


Figure 7.5. Print preview

References

- [1] Belton, V. (1990), "Multiple Criteria Decision Analysis - Practically the Only Way to Choose", L.C. Hendry and R.W. Englese (eds.), *Operational Research Tutorial Papers*, Birmingham: *Operational Research Society*, 53-101.
- [2] Bratley, P., Fox, B.L., Schrage, E.L. (1983), *A Guide to Simulation*, Springer-Verlag, New York, 180-213.
- [3] Brownlow, S.A. and Watson S.R. (1987), "Structuring Multi-attribute Value Hierarchies", *Journal of the Operational Research Society* **38**, 309-317.
- [4] Farquhar, P.H. (1984), "Utility Assessment Methods", *Management Science* **30**, 1283-1300.
- [5] Gallego, E., Ríos-Insua, S., Mateos, A., Ríos Insua, D. (1998), "Contaminated Aquatic Ecosystems Restoration: A Case Study", F. J. Girón (ed.), *Applied Decision Analysis*, Kluwer, Boston, 19-34.
- [6] Hershey, J.C., Kunreuther, H.C., Schoemaker, P.J. (1982), "Sources of Bias in Assessment Procedures for Utility Functions", *Management Science* **28**, 936-953.
- [7] Jaffray, J.Y. (1989), "Some Experimental Findings on Decision Making Under Risk and Their Implications", *European Journal of Operational Research* **38**, 301-306.
- [8] Jiménez, A., Ríos-Insua, S. and Mateos, A. (2002), "A Decision Support System for Multiattribute Utility Evaluation Based on Imprecise Assignments", *Decision Support Systems* **36**, 1, 65-79.
- [9] Jiménez, A., Ríos-Insua, S. and Mateos, A. (2003), "Monte-Carlo Simulation Techniques in a Multi-Attribute Decision Support System", *Proceedings of the 12th IASTED International Conference on Applied Simulation and Modelling*, ACTA Press, 85-90.
- [10] Keeney, R.L. and Raiffa H. (1976), *Decisions with Multiple Objectives: Preferences and Value-Tradeoffs*, Wiley, New York.
- [11] Kirkwood, C.W. (1997), *Strategic Decision Making. Multiobjective Decision Analysis with Spreadsheets*, Duxbury Press, Belmont.

- [12] Mateos, A., Jiménez, A. and Ríos-Insua, S. (2003), "Solving Dominance and Potential Optimality in Imprecise Multi-Attribute Additive Problems", *Journal of Reliability Engineering and System Safety* **79**, 2, pp. 253-262.
- [13] McCord, M. and de Neufville, R. (1986), "Lottery Equivalents: Reduction of the Certainty Effect Problem in Utility Assessment", *Management Science* **32**, 56-61.
- [14] Raiffa, H. (1982), *The Art and Science of Negotiation*, Harvard University Press: Cambridge, Mass.
- [15] Ríos, S., Ríos-Insua, S., Ríos Insua, D. and Pachón, J.G. (1994), "Experiments in Robust Decision Making", Ríos, S. (ed.), *Decision Theory and Decision Analysis: Trends and Challenges*, Kluwer, Boston, 233-242.
- [16] Ríos Insua, D., Gallego, E., Mateos, A. and Ríos-Insua, S. (2000), "MOIRA: A Decision Support System for Decision Making on Aquatic Ecosystem Contaminated by Radioactive Fallout", *Annals of Operations Research* **95**, 341-364.
- [17] Ríos Insua, D. and French, D. (1991), "A FrameWork for Sensitivity Analysis in Discrete Multi-Objective Decision-Making", *European Journal of Operational Research* **54**, 176-190.
- [18] Ríos Insua, D. (1990), *Sensitivity Analysis in Multiobjective Decision Making*, LNEMS 347, Springer, Berlin.
- [19] Stewart, T. J. (1996), "Robustness of Additive Value Function Method in MCDM", *Journal of Multi-Criteria Decision Analysis* **5**, 301-309.
- [20] von Nitzsch, R. and Weber, M. (1988), "Utility Function Assessment on a Micro-computer: An Interactive Procedure", *Annals of Operations Research* **16**, 149-160.